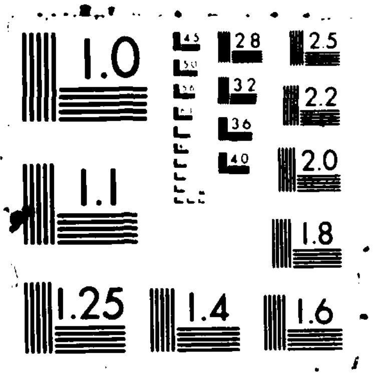
AD-A182 591 ROBOTICS APPLICATIONS FOR THE TESTING OF INERTIAL SENSORS(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING J Y GREIG MAY 87 1/1

UNCLASSIFIED AFIT/GE/ENG/87J-4 F/G 14/2 NL

END
8 87
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0

DTIC FILE COPY

AD-A182 591

ROBOTICS APPLICATIONS FOR THE TESTING

OF INERTIAL SENSORS

THESIS

Joy Y. Greig

AFIT/GE/ENG/87J-4

DTIC
ELECTE
JUL 2 7 1987
S D
E

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

This document has been approved for public release and sale; its distribution is unlimited.

87 7 22 0 4

1

AFIT/GE/ENG/87J-4

ROBOTICS APPLICATIONS FOR THE TESTING OF INERTIAL SENSORS

THESIS

Joy Y. Greig

AFIT/GE/ENG/87J-4

DTIC
ELECTE
JUL 2 7 1987
S D
E

Approved for public release; distribution unlimited

AFIT/GE/ENG/87J-4

ROBOTICS APPLICATIONS FOR THE TESTING OF INERTIAL SENSORS

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science

Accession For
NTIS GRA&I ☒
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A-1

Joy Y. Greig, B.A.

June 1987

Approved for public release; distribution unlimited

Acknowledgements

There were many different people who at some point contributed to the completion of this thesis. I wish to give special thanks to Col. Dan Biezad, my advisor, for his encouragement and invaluable direction of my thinking at the times I needed it most. Thanks, too, to the members of my committee, Dr. Matthew Kabrisky, Col. Z. Lewantowicz, and Capt. Tom Clifford, for their suggestions and help.

In addition to my committee, Dr. Gary Lamont, Dr. Dan Reynolds, and Dr. Joseph Cain provided expert guidance for the economic and statistical aspects of the study. I also received willing and timely assistance from the staff of the engineering school, for which I am truly grateful.

And finally, I must give thanks to those who supported me most at the times when things were the least encouraging: my terrific daughters Monica and Audra who for two years missed out on having all the special time with Mom that they were used to, and my friends and family who prayed for, loved and encouraged me into doing my best.

Table of Contents

Page

Acknowledgements ii

List of Figures vi

List of Tables vii

Abstract viii

I. Introduction 1

II. Technical Feasibility 4

- Vertical-Seeking Test 5
  - Theory 5
  - Implementation 7
  - Results 11
  - Summary 13
- Accelerometer Fourpoint Test 15
  - Theory 15
  - Experiment Setup and Methodology 16
  - Results 18
- Gyroscope Step-Tumble Test 21
  - Theory 21
  - Experiment Setup and Methodology 23
  - Results 25
- Robot Performance Criteria 27
  - Determination of Performance Criteria 27
  - Identification of Suitable Robots 29
- Summary 31

III. Economic Feasibility 32

- Economic Analysis Criteria 32
- Life Cycle Costing 33

Life Span Costs 33
Research and Development Costs 34
Investment Costs 35
Operating Costs 36

Results 38

IV. Limitations 39

Practical Engineering Limits 39

Flexibility of the Robot Arm 39
Control Schemes 41

Computer Modeling Limits 42

Applications 42
Mechanical Model 42
Dynamics Model 43

Measurement and Instrumentation Limits 43

Summary 44

V. Conclusions and Recommendations 45

Appendix A. Robot Safety Summary 48

General Procedures 48
Specific Precautions 48
Emergency Shutdown 49

Appendix B. Software for Vertical-Seeking Test 50

Overview 50

Program Descriptions 50

GRAVITY.MAIN 50
GRAVITY.MAX 51
DELAY 52

VAL II Program Listings 54

Appendix C. Accelerometer Fourpoint Test 56

Theory 56
Software 57

Appendix D. VAL II Gyro Programs and Output Listings 61

Program Listings 62

        STTEST.CP 62
        DELAY.SP 62
        STTEST1.SP 63
        STTEST2.SP 63
        SCALFACT.TEST. 64

    Sample Output Listings Using A to D Converter 65

Appendix E. Overview of ROBSIM 67

    Synthesis (INITDRVR) 67
    Simulation (SIMDRVR) 68
    Analysis (POSTDRVR) 69
    ROBSIM Installation 70
    Summary 70

Appendix F. Robot Kinematics and Inertial Navigation 71

    Robot Kinematics. 71
    Inertial Navigation Kinematics 72
    Unification of Notation 73

    Notation and Tutorial 73

        Notation 74
        Homogeneous Transformation Matrices 75
        Homogeneous Transformations 78
        Quaternions 82

Bibliography 85

Vita 88

List of Figures

Figure Page

2.1. PUMA 560 Robot Arm 8

2.2. Definition of Euler Angles O, A, and T 10

2.3. Tool Coordinate Frame of the PUMA 560 17

2.4. Spreadsheet of Robotic and Non-Robotic Testing Units 30

4.1. ROBSIM Output for the PUMA 560 Shoulder Rotation 41

B.1. READY Position of PUMA 560 51

B.2. Flowchart for Subroutine GRAVITY.MAX 53

F.2. Coordinate Frames of Serial Manipulator 79

F.3. Homogeneous Displacement Matrix 82

F.4. Quaternion Representation 84

List of Tables

Table Page

2.1 Position and Orientation of Points V_1 and V_2 12

2.2 Accelerometer Performance Characteristics from Fourpoint Tests on Robot Arm 19

2.3 Sample Accelerometer Performance Characteristics from Fourpoint Tests on Vertical Table 20

2.4 Relationship of Gyro Drift Coefficients to Fourier Coefficients of OA|EA Tumble Test Data 23

2.5 Hardware Used to Conduct Gyro Testing 24

2.6 Performance Model Equation Coefficients 26

3.1 Performance Characteristics and Base Prices of Robotic and Non-robotic Testing Units 32

3.2 Examples of Life Cycle Costing Worksheet 36

3.3 Total Life Cycle Costs 37

4.1. Rotation Alignment Errors 40

D.1 Fourier Coefficients from Step-Tumble Test 66

Abstract

This thesis investigates the technical and economic feasibility of using robots as testbeds for inertial sensors in lieu of the current generation of unique, expensive, and relatively inflexible gyro and accelerometer test facilities.

A PUMA 560 robot arm is used as the experimental testbed. The design and demonstration of three tests are described which illustrate the alignment, calibration, and resulting performance of the robot as a test device. Actual high precision inertial accelerometers and gyros were used to establish a performance baseline and to evaluate the robot's test capabilities.

The robot was programmed to act as a precision test servomechanism which was calibrated and aligned automatically using the high quality sensors. The robot was effective in providing quick-look results of error coefficient parameters for bias, scale factor, and elastic (g-squared) effects, but as expected could not provide inherently the precision required for very high quality sensor testing.

Testing was easily implementable and varied to suit individual applications, and this illustrated the value and potential for devising new tests which cannot be performed on existing precision test equipment. The problem of lack of precision was investigated using an advanced computer simulation. This simulation shows that there are serious limitation due to unmodeled noise and flexure of the robot arm, which is significant for the high precision required for inertial testing. The need for this precision does not have to be supplied by

the robot, however, if sufficiently precise measurement tools (lasers) are used to establish reference position and attitudes. Economic analyses established that, given precision measurement capability, using a robot arm as a test mechanism is viable, cost effective, and a practical engineering test procedure.

I. Introduction

Specialized test facilities, such as the Central Inertial Guidance Test Facility (CIGTF) at Holloman Air Force Base, New Mexico, are responsible for the testing of high quality inertial rate sensors and accelerometers. Due to the large investment in resources, it is important that all sensors be free from major defects when scheduled for precision testing. Initial sensor checkout tests, for example, should not tie up unique and specialized test equipment which may cost millions of dollars (2).

Although these expensive devices for testing inertial sensors have been very effective, due to their unique design they often lack the flexibility required to implement new test procedures. Moreover, there is little evidence of rapid innovation in designing and building new test fixtures with enhanced capabilities. These problems of cost, inflexibility, and lack of new capabilities are serious ones. A potential approach to addressing these problems comes from the rapidly developing engineering science of robotics, where cost is decreasing due to the exponential rise in the number of units being produced (increasing from 20,000 units in 1976 to 250,000 in 1984), and the digital capabilities being designed into robots have the potential to provide flexibility in systems tests and data acquisition (23). Finally, robotics is a highly innovative area fueled by vast research funding. It is probable that if the key difficulty of precision can be solved, the use of programmable robots for inertial testing should become a reality.

This thesis investigates the feasibility of robotics applications to inertial component testing by addressing three major areas: technical feasibility, economic feasibility, and limitations.

Technical feasibility is discussed in Chapte~ II where the design and implementation of three tests using a PUMA 560 robot arm (30;31) are accomplished. The tests are a vertical-seeking test for robot arm alignment, an accelerometer fourpoint test, and a gyroscope step-tumble test.

The PUMA 560 was suitable for the tests since it met the basic testing requirements: six degrees of freedom, greater than 360 degrees of rotation in the wrist area, repeatability of at least 0.008 inches, and off-line programming.

The first demonstration is a method of locating local vertical using a high-precision accelerometer. Because of the robot's maneuverability and input/output capability, it can be programmed to respond to outputs of an accelerometer and thus align a robot arm for check-out tests on sensors. There are other instruments available to aid in aligning the arm further for testing the precision accelerometers themselves (19).

Standard static tests for determining the error coefficients of an accelerometer and a gyro follow. A standard fourpoint test for accelerometers (33) and a standard step-tumble test for rate gyros (12;34) are designed using the robot arm as the test device.

Description of the robotic implementation of the tests and data analysis are included. The object of the demonstrations was to show the adaptability of the robot arm and the feasibility of programming and performing the standard tests.

Finally in Chapter II robot performance criteria for supporting inertial sensor testing are developed. Current industrial robots which meet the criteria are located (29), and four are selected for study and comparison with three non-robotic precision test units.

Chapter III contains an investigation of economic feasibility where simple life cycle costs for robots are defined and compared with the non-robotic units of Chapter II (15;1).

Current robotics limitations are discussed in Chapter IV, and the attempt to solve the precision problem using computer simulations is illustrated by a case study. The robot simulator used is the Integrated Robotic System Simluation Program ROBSIM developed by Martin Marietta Denver for NASA Langley (13).

Chapter V examines the potentials of robotics for precision sensing, cost reduction, and development and application of new test technologies. Recommendations are made for further research and development of robotics applications to inertial sensor testing.

II. Technical Feasibility

The robot in itself is not a precision device relative to inertial sensor accuracies required for testing. These accuracies were investigated in this study to determine the feasibility of using a robot as a testbed. Three tests on a PUMA 560 robot arm were accomplished to illustrate this and to examine robot performance criteria for sensor/system laboratory testing.

A PUMA 560 robot arm with a VAL II™ (Versatile Assembly Language) operating system was available for use in the research (30). A Systron-Donner 4841F high-precision accelerometer (0.0001 μ-g rms error) and a Humphrey Model RG51-0106-1 rate gyro were the test sensors. Arm motion programs in VAL II placed the sensors at the appropriate positions and performed the rotations for the tests.

As with any other test stand, a robot must be calibrated and aligned. To demonstrate the alignment of the robot arm with local vertical, a vertical-seeking test was designed, using the output of the high-precision accelerometer and the PUMA 560's operating system to accomplish the calibration.

Next two standard inertial sensor laboratory tests were adapted to the robot testbed to determine the practicality of using a robot arm for static sensor testing and the robot performance criteria for the testbed application. A standard accelerometer fourpoint test was performed and the data analyzed to investigate the precision capabilities of the arm. A standard gyroscope step-tumble test was then implemented to demonstrate reconfiguration flexibility of the arm.

Vertical-Seeking Test

To demonstrate calibration and alignment of the robot arm, a vertical-seeking test was performed. In this test the output of the Systron-Donner 4841F accelerometer to the robot operating system was used to direct the arm in locating local vertical. In an actual testing situation a high-precision accelerometer, a triad of accelerometers, a laser, or some other means could be used either to verify the robot's position or to position it if its own positioning system were limited. (Robot positioning accuracies are discussed in Chapter IV.) In this demonstration, however, a single accelerometer was used to locate local vertical.

The direction of vertical could be determined by simply maximizing a single accelerometer reading and using a numerical algorithm to zero in on vertical. However, most practical applications are faced with limited numerical accuracy in reading an accelerometer; because of the non-linear nature of accelerometer reading accuracies, it is more accurate to find horizontal. The following sections discuss the use of a single accelerometer used in conjunction with the geometric configuration of the PUMA 560 robot to determine the direction of vertical. (This section is derived from work by the author and Drew A. Karnick.)

Theory. A Systron-Donner 4841F accelerometer is available for implementation with a PUMA 560 robot manipulator in order to formulate the direction of gravity. The output of the accelerometer is a voltage between -10 and +10 volts that corresponds linearly to the accelerometer input of -1 to +1 g's. Available to read the accelerometer is an analog to digital converter that transforms the input signal to an integer ranging in value from -2048 to 2047. The user reads the voltage by scaling this number. For example if the number read is 1000, the voltage is calculated as:

$$V = 10 \text{ Volts} \times (1000/2047) = 4.88 \text{ volts} \qquad (2.1)$$

However, the reading is of limited accuracy. Greater accuracy could be obtained using a 16-bit A-to-D converter, but one was not available for for this research.

With the 12-bit converter, if the reading is positive the accuracy is:

$$\text{accuracy} = \pm\ 10/2047 = \pm\ 0.004885 \text{ volts} \qquad (2.2)$$

This error in reading translates into significantly different measurement errors when determining the direction of gravity. An accelerometer reading is essentially:

$$\text{Acc. Reading} = \cos(\theta) \times g \qquad (2.3)$$

where g is the magnitude of the gravity vector (measured in volts) and θ is the angle between the accelerometer input axis and the gravity vector.

If the accelerometer is used to determine vertical and is reading exactly 10 volts, the actual reading is in the range from 9.995 volts to 10.005 volts. If it is assumed that the worst has occurred and that the actual reading is 9.995 volts, the angle between vertical and the accelerometer can be calculated as:

$$\begin{aligned}\theta &= \cos^{-1}(\text{reading}/g) = \cos^{-1}(9.995/10)\\ &= 1.81 \text{ degrees} \qquad (2.4)\end{aligned}$$

If the same voltage error evaluated in Equation (2.4) occurs at θ = 0, the reading is:

$$\theta = \cos^{-1}(\text{reading}/g) = \cos^{-1}(0.005/10)$$
$$= 89.97 \text{ degrees} \tag{2.5}$$

This translates into an error of only 0.03 degrees, more than an order of magnitude increase in performance! To achieve the same performance by finding vertical would require a large increase in voltage reading accuracy. Substituting 0.03 degrees into Equation (2.3) yields:

$$\text{accel. reading} = \cos(0.03) \times 10 \text{ volts}$$
$$= 9.9999986 \text{ volts} \tag{2.6}$$

The conclusion to be drawn is that when using an accelerometer to find the direction of gravity and faced with limited accuracy in accelerometer readings, it is more accurate to find the minimum of this acceleration by finding horizontal. However, one basic fact must not be overlooked: a single determination of vertical allows (with some error) the finding of vertical, while a single determination of horizontal only defines a vector that lies in a plane that is at an angle of 90 degrees from the gravity vector. A second distinct determination of horizontal is required to define this plane, thereby finding the direction of gravity. In addition to being distinct, to minimize a loss in numerical precision the second horizontal vector should be at an angle of 90 degrees with the first horizontal vector.

Implementation. The natural precision geometry of the PUMA 560 manipulator (see Figure 2.1) supplies the proper configuration to determine vertical. Since horizontal is determined twice, it is desirable to obtain two determinations of vertical about perpendicular axes of rotation and implement this for joints with the largest degrees of rotational freedom. It is also desirable to minimize the actuator output required to hold the arm in position as well as the horizontal extension of the arm, as they may lead to errors in determining the orientation of the accelerometer in relation to the PUMA 560 (16). These criteria lead to the natural choice of Joint 5 (wrist bend) in conjunction with a 90 degree rotation in Joint 1 (waist); see Figure 2.1.

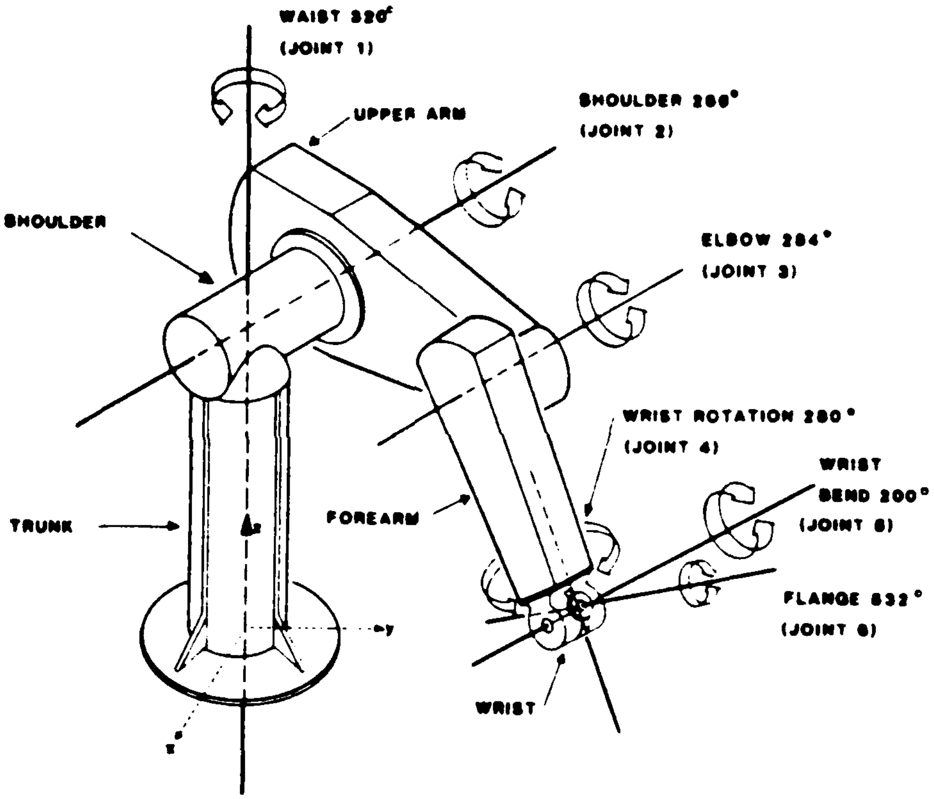


Figure 2.1. PUMA 560 Robot Arm (Reference 30)

At each determination of horizontal the orientation and position of the accelerometer is recorded. This allows the formulation of two vectors which define a plane perpendicular to the direction of vertical. The cross product of the these two vectors defines the direction of vertical; the tool is then oriented to point in the direction of vertical.

The first horizontal vector is determined with the PUMA 560 in the READY position, where the axis of rotation of Joint 5 is parallel to the World y-axis (the World Coordinate System is the base coordinate system indicated in Figure 2.1). It is assumed that the input axis of the accelerometer lies in the World x-z plane. This assumption allows the determination of the direction of the input axis of the accelerometer when it has been oriented in the direction of vertical by decomposition of the position of the tool. Decomposition of a point yields the following information:

X Y Z O A T

where X, Y, and Z define the position of the tool in World coordinates and O, A, and T are angles which define the orientation of the tool (Figure 2.2). The variable A defines the angles between the tool z-axis and the x-y plane.

The tool z-axis is horizontal and lies along the unit vector

$$V_1 = \cos A_1 \cdot i + 0 \cdot j + \sin A_1 \cdot k \tag{2.7}$$

which is defined in the World Coordinate System.

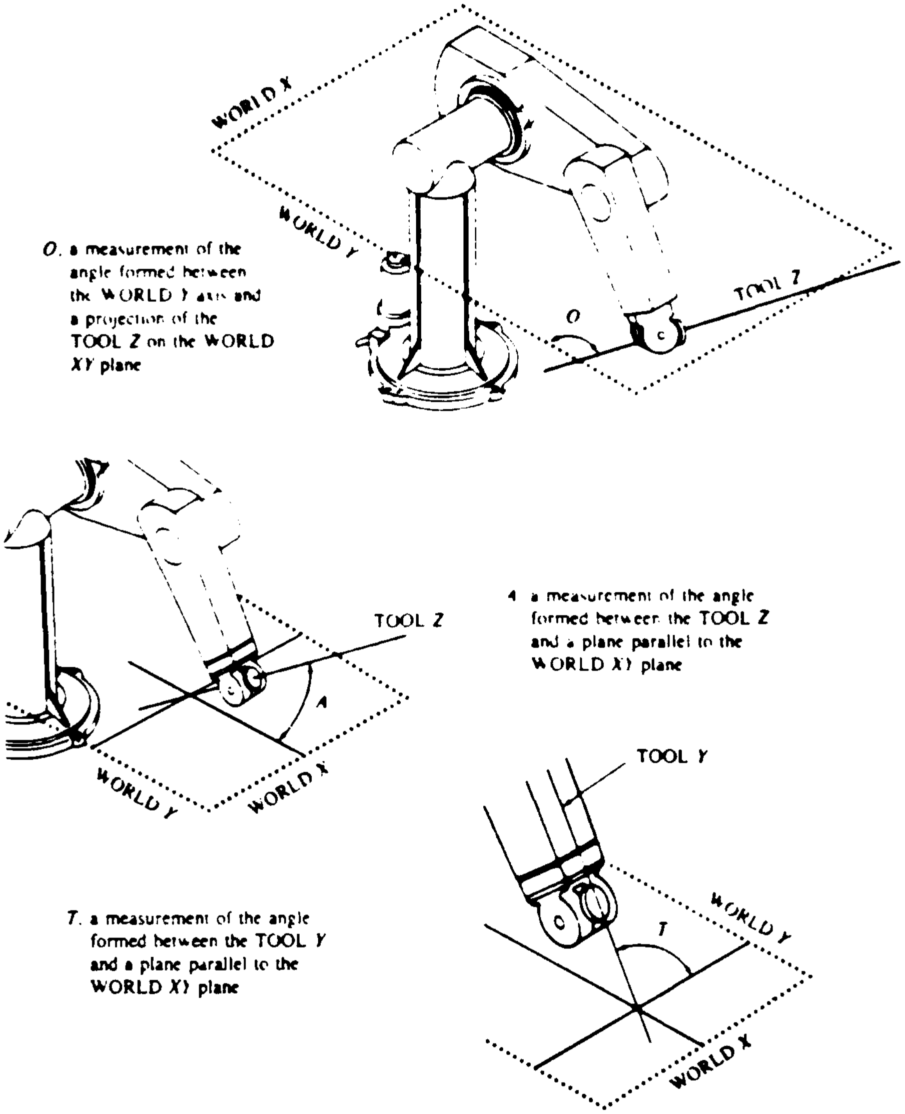
The second horizontal vector is then determined by placing the rotational axis of Joint 5 parallel with the World x-axis. The tool axis is positioned horizontally by rotation about the Joint 5 axis; it is assumed that the accelerometer input axis lies in the World y-z plane. The tool z-axis then lies along the unit vector:

$$V_2 = 0 \cdot i + \cos A_2 \cdot j + \sin A_2 \cdot k \qquad (2.8)$$

which is defined in the World Coordinate System.

The cross product of the vectors V_1 and V_2 (a vector oriented along vertical) can be defined as the determinant of the following matrix:

$$V_3 = \begin{vmatrix} i & j & k \\ \cdots & V_1 & \cdots \\ \cdots & V_2 & \cdots \end{vmatrix} \qquad (2.9)$$



O. a measurement of the angle formed between the WORLD *Y* axis and a projection of the TOOL *Z* on the WORLD *XY* plane

A. a measurement of the angle formed between the TOOL *Z* and a plane parallel to the WORLD *XY* plane

T. a measurement of the angle formed between the TOOL *Y* and a plane parallel to the WORLD *XY* plane

Figure 2.2. Definition of Euler Angles O, A, and T (Reference 31)

Substituting for Equations (2.7) and (2.8) yields:

$$V_3 = \begin{vmatrix} i & j & k \\ \cos A_1 & 0 & \sin A_1 \\ 0 & \cos A_2 & \sin A_2 \end{vmatrix} \qquad (2.10)$$

$$V_3 = -\cos A_2 * \sin A_1 \cdot i - \cos A_1 * \sin A_2 \cdot j + \cos A_2 \cos A_1 \cdot k \qquad (2.11)$$

This vector contains the information needed to position the tool z-axis along the calculated direction of vertical. An angle θ_1 is defined as the arctangent of the World x and y components of Equation (2.11):

$$\theta_1 = \tan^{-1}\left[\frac{\cos A_1 * \sin A_2}{\cos A_2 * \sin A_1}\right] \qquad (2.12)$$

An angle θ_2 is defined as the angle between the world z component of Equation (2.11) and the World x-y plane:

$$\theta_2 = \tan^{-1}\left[\frac{\mathrm{SQRT}[(\cos A_1 * \sin A_2)^2 + (\cos A_2 * \sin A_1)^2]}{\cos A_2 * \cos A_1}\right] \qquad (2.13)$$

Note that the sign of the numerator and denominator is important for uniquely defining an angle. The arctangent function in the VAL II programming language requires the input of the numerator and denominator separately.

The angle θ_1 can be used to define the O angle in the world x-y plane while the θ_2 angle is used to define the A angle. The PUMA 560 is then oriented in this direction.

Results. The algorithms previously discussed are implemented on the PUMA 560 using the Unimate controller and VAL II programming language. Appendix A contains a brief summary of the safety aspect of operating the PUMA 560. Appendix B contains a listing and description of all programs; this section discusses only the outputs of these simulations. The simulation produced two points V_1 and V_2. These two points are defined in Table 2.1.

Table 2.1

Position and Orientation of Points V_1 and V_2

Point	X*	Y	Z	O**	A	T
V1	36.00	149.09	863.81	90.006	1.071	0.000
V2	-149.09	36.03	863.97	-180.000	0.917	-0.011

* X, Y, and Z are in millimeters
** O, A, and T are in degrees

The cross product of these two vectors is defined as (see Equations (2.9) and (2.10)):

$$V_3 = \begin{vmatrix} i & j & k \\ \cos(1.071) & 0 & \sin(1.071) \\ 0 & \cos(.917) & \sin(.917) \end{vmatrix} \tag{2.14}$$

$$V_3 = -0.0187 \cdot i - 0.0160 \cdot j + 0.9997 \cdot k \tag{2.15}$$

The angles θ_1 and θ_2 are then calculated from Equations (2.12) and (2.13):

$$\theta_1 = \tan^{-1}\left[\frac{-0.0160}{-0.0187}\right] = 220.55^\circ \tag{2.16}$$

$$\theta_2 = \tan^{-1}\left[\frac{0.0246}{0.9997}\right] = 1.406^\circ \tag{2.17}$$

The PUMA 560 is then oriented according to these two angles. The θ_1 angle corresponds to the O angle discussed previously while the θ_2 angle corresponds to the A angle discussed previously.

Since the Systron-Donner 4841F accelerometer is the most accurate instrument available, analysis is limited to that of a qualitative nature. First, visual inspection can ascertain whether the tool is orientated in the direction of vertical. Visual inspection of the orientation of the tool did indicate that the vertical-seeking algorithm found vertical.

Secondly, one would expect the z axis of the World Coordinate System to be roughly aligned with vertical; therefore, the cross product of the two vectors situated in the horizontal plane results in a vector that has its major component along the World z axis. Equation (2.15) clearly shows that this is the case.

The theory and analysis presented here have presumed no robot joint positioning errors. There are, however, small accumulated errors via quantization of robot movement and calculations by the robot arm controller (30). No attempt was made to include these errors in the vertical-seeking algorithm. The algorithm did, however, locate vertical more precisely than could be done by simply placing the arm in the "ready" position, or by using a single accelerometer output determination.

Summary

The development and implementation of an algorithm to align and calibrate the robot arm for use as a testbed has been discussed in this section. A precision accelerometer and the robot's controller system were used to determine local vertical. The reasons for finding horizontal rather than vertical for this method have been presented and the algorithm shown to be successful in finding vertical more accurately than a direct measurement of vertical using a single accelerometer.

Accelerometer Fourpoint Test

The object of the fourpoint test was to investigate the degree of testing precision achievable with the PUMA 560 Robot Arm. A series of fourpoint tests was performed on the Systron-Donner 4841F accelerometer. The accelerometer output was analyzed by calculating and determining the stability of the accelerometer scale factor, 1-g bias, null bias, and misalignment angle.

Theory. The complete performance-model equation for an accelerometer is in Appendix C. When an accelerometer is rotated about its output axis, the accelerations about that axis become zero, reducing the model to (32:C-4):

$$A_{ind} = E/K_1 = K_0 + a_i + K_2 a_i^2 + K_3 a_i^3 + \delta_0 a_p + K_{pp} a_p^2 + K_{ppp} a_p^3 \qquad (2.18)$$

where

a_i = Acceleration component along input axis (g)

a_p = Acceleration component along pendulous axis (g)

E = Accelerometer output (output units)

A_{ind} = Indicated acceleration (g)

K_0 = Bias (g)

K_1 = Scale factor (output units/g)

K_2 = Second order non-linearity coefficient (g/g^2)

K_3 = Third order non-linearity coefficient (g/g^2)

δ_0 = Input axis misalignment about output axis (radians)

K_{pp} = Second-order cross axis coefficient (g/g^2)

K_{ppp} = Third-order cross axis coefficient (g/g^3).

The performance-model equation at the four positions of 90°, 270°, 0°, and 180° becomes (32:C-4):

a. At the 90° position, $a_i = +1g$ and $a_p = 0g$, so

$$E(90^\circ) = (K_0 + 1 + K_2 + K_3)K_1 . \qquad (2.19)$$

b. At the 270° position, $a_i = -1g$ and $a_p = 0g$, so

$$E(270^\circ) = (K_0 - 1 + K_2 - K_3)K_1 . \qquad (2.20)$$

c. At the 0° position, $a_i = 0g$ and $a_p = +1g$, so

$$E(0^\circ) = (K_0 + \delta_0 + K_{pp} + K_{ppp})K_1 . \qquad (2.21)$$

d. At the 180° position, $a_i = 0g$ and $a_p = -1g$, so

$$E(180^\circ) = (K_0 - \delta_0 + K_{pp} - K_{ppp})K_1 . \qquad (2.22)$$

The +1g (90°) and -1g (270°) positions of the accelerometer are then used to determine the 1g bias (M_0) and the two-point scale factor (M_1). The null positions (0° and 180°) yield the accelerometer null bias (N_0) and misalignment error (δ_0) (33:A-3). These characteristics are calculated using the following relationships:

$$M_1 = K_1(1 + K_3) = \tfrac{1}{2}[E(90^\circ) - E(270^\circ)] \quad \text{(output units/g)} \qquad (2.23)$$

$$M_0 = K_0 + K_2 = \{[E(90^\circ) + E(270^\circ)]\ 10^6\}/2M_1 \quad (\mu g) \qquad (2.24)$$

$$N_0 = K_0 + K_{pp} = \{[E(0^\circ) + E(180^\circ)]\ 10^6\}/2M_1 \quad (\mu g) \qquad (2.25)$$

$$\delta_0' = \delta_0 + K_{ppp} = \{[E(0^\circ) - E(180^\circ)](2.06 \times 10^5)\}/2M_1 \quad \text{(arcsec)}. \qquad (2.26)$$

Experiment Setup and Methodology.

Hardware. The Systron-Donner 4841F accelerometer is a conventional single-axis, pendulous, fluid floated, torque rebalance accelerometer, with an analog output in volts direct current (VDC) proportional to the applied acceleration. Input voltages of +15 VDC and -15 VDC were supplied by two Datel Voltage Calibrators, Model DCV8500. A 6.8 kΩ precision resistor was used to scale the output to 1 volt/g. The output was measured by a Dana Digital Multimeter Model 5900.

Procedure. For the series of fourpoint tests, the accelerometer was secured to an aluminum mount which was screwed on to the robot tool flange (see Figure 2.1). The robot arm was aligned parallel to local gravity. The pendulous axis (PA) of the accelerometer was aligned parallel to the Y-axis of the tool flange (see Figure 2.3) and its input axis (IA) perpendicular to the Y-axis of the tool flange. The robot wrist joint was rotated 90°, followed by a 90° rotation of Joint 5, in order to position the accelerometer IA up and parallel to local vertical.

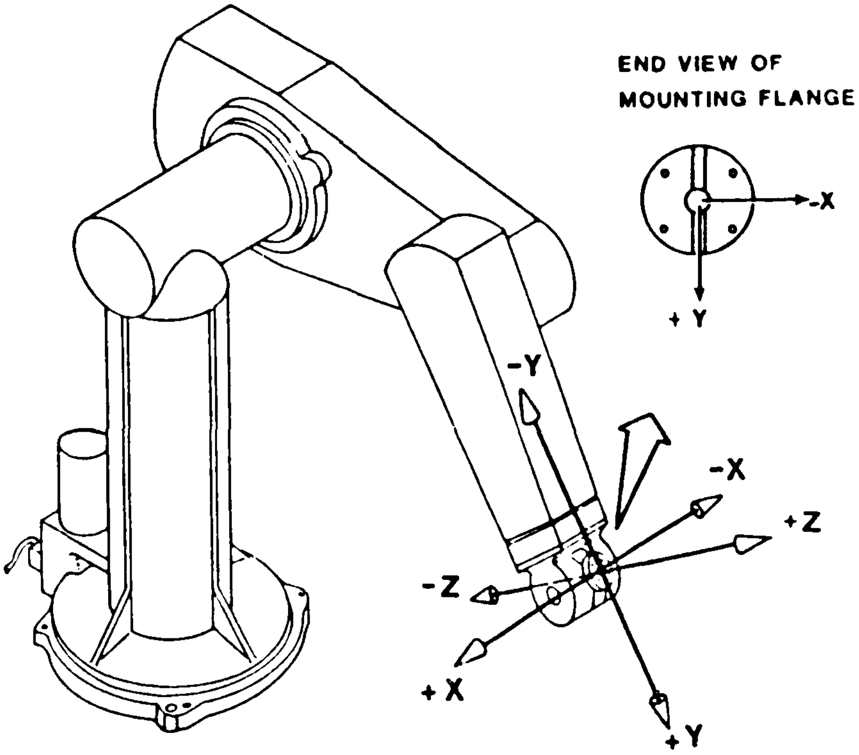


Figure 2.3. Tool Coordinate Frame of the PUMA 560

The flange was then rotated in the following pattern (33:A-3):

a. Initial position at 90° (IA up)

b. Rotate clockwise (CW) to 270° (IA down)

c. Rotate counterclockwise (CCW) to 180° (IA horizontal-null)

d. Rotate CW to 0° (IA horizontal null)

e. Return to 90°.

Twenty fourpoint tests were performed with the robot pausing for 15 seconds at each position. The test engineer read and recorded the voltage manually.

Software Development. The software was designed to rotate the accelerometer to the four positions as previously discussed and allow sufficient time to read the accelerometer output voltage at those positions.

This was accomplished by inputting the motion commands with the VAL II operating system. The accelerometer was rotated to the four positions by using the DRIVE command to rotate Joint 6 (the flange) the appropriate number of degrees.

The program is a simple main program with a DO loop implemented to command a user-defined number of fourpoint tests to be run per test session and remain a user-defined time at each position. The DELAY subprogram allows the time in position to be varied according to user choice. The FOUR (fourpoint) and DELAY programs and listing of an execution of the program are in Appendix C.

Results. The results of the fourpoint test are summarized in the following table. Although the performance characteristic values are larger than those derived from fourpoint tests of similar instruments (see Table 2.3 from 33:27-28), the standard deviations and peak-to-peak spread are comparable. The laboratory environment for this research was much less controlled than that of a test facility such as CIGTF; noise sources from the laboratory and perhaps from the robot arm itself, and lack of temperature control contributed to the magnitude of the coefficients. However, the stability of the outputs is an indication of the positioning repeatability of the robot arm.

The goal of the fourpoint tests was to investigate degree of testing precision achievable with the PUMA 560 Robot Arm. The data showed that positioning precision can be achieved. This demonstrates that the robot is certainly a viable testbed for performing initial performance checks on a high-accuracy sensor, and perhaps the evaluation tests as well on lower-accuracy instruments. A more controlled test environment and an evaluation of the noise characteristics of the robot arm are necessary to completely determine its potential for evaluation tests of high-accuracy sensors.

Table 2.2

Accelerometer Performance Characteristics
from Fourpoint Tests on Robot Arm

	Scale Factor (volts/g)	1-g Bias (μg)	Null Bias (μg)	Misalignment Angle (arcsec)
Mean	1.018805	1207	1720	8154
Standard Deviation	29.45 ppm	60	66	9
Peak-to-peak Variation	0.000115	241	255	30

Table 2.3

Sample Accelerometer Performance Characteristics from Fourpoint Tests on Vertical Table (Reference 33:27)

	Scale Factor (volts/g)	1-g Bias (μg)	Null Bias (μg)	Misalignment Angle (arcsec)
Mean	0.024934	184.5	148.4	-30.6
Standard Deviation	40 ppm	45.8	36.4	7.5
*Peak-to-peak Variation	471 ppm	471	244	58

*Over 39 days. No data available for a single day's testing.

Gyroscope Step-Tumble Test

The purpose of the gyroscope (gyro) step-tumble test was to demonstrate the maneuverability of a robot arm and the ease of reconfiguring the robot for different tests. For the step-tumble test the robot must be positioned to align the gyro's output axis parallel to the earth's rotational axis pointing north and then pointing south. The output of the gyro in these orientations is used to calculate the gyro drift characteristics.

Theory. The assumed performance model for a single degree of freedom (SDOF) gyro is:

$$S_T Gi = D_F + D_I a_I + D_O a_O + D_S a_S + D_{II} a_I^2 + D_{SS} a_S^2$$
$$+ D_{IO} s_I a_O + D_{IS} a_I a_S + D_{OS} a_O a_S + w_I \qquad (2.27)$$

where

i = current flow through the torque generator (ma)

S_{TG} = sensitivity of the torque generator (deg/hr/ma)

a_I = acceleration along the gyro IA (g)

a_O = acceleration along the gyro OA (g)

a_S = acceleration along the gyro SA (g)

D_F = gyro drift which is insensitive to acceleration (deg/hr)

$D_I a_I$ = gyro drift (deg/hr) attributable to acceleration along the IA, where D_I (deg/hr/g) is a drift coefficient

$D_O a_O$ = gyro drift (deg/hr) attributable to acceleration along the OA, where D_O (deg/hr/g) is a drift coefficient

$D_S a_S$ = gyro drift (deg/hr) attributable to acceleration along the SA, where D_S (deg/hr/g) is a drift coefficient

$D_{II} a_I^2$ = gyro drift (deg/hr) attributable to the square of acceleration along the IA, where D_{II} (deg/hr/g^2) is a drift coefficient

$D_{SS}a_S^2$ = gyro drift (deg/hr) attributable to the square of acceleration along the SA, where D_{SS} (deg/hr/g^2) is a drift coefficient

$D_{IO}a_Ia_O$ = gyro drift (deg/hr) attributable to the product of acceleration along the IA and OA, where D_{IO} (deg/hr/g^2) is a drift coefficient

$D_{IS}a_Ia_S$ = gyro drift (deg/hr) attributable to the product of acceleration along the IA and SA, where D_{IS} (deg/hr/g^2) is a drift coefficient

$D_{OS}a_Oa_S$ = gyro drift (deg/hr) attributable to the product of acceleration along the OA and SA, where D_{OS} (deg/hr/g^2) is a drift coefficient

w_I = angular velocity, in inertial space, of the gyro case about the IA (deg/hr)

g = local acceleration of gravity, defined positive upward (cm/sec^2)

The magnitude of the variable quantities (a_I, a_S^2, a_O, a_S, etc.) in the assumed performance model are dependent on gyro orientation relative to the local gravity vector and any other applied accelerations (12).

In order to calculate the drift coefficients, the voltage output of the gyro is first least squares fit to the model:

$$\text{Voltage Output} = C_0 + C_1\sin A_{TA} + C_2\cos A_{TA} + C_3\sin 2A_{TA} + C_4\cos 2A_{TA} \qquad (2.28)$$

where

C_i = Fourier coefficients

A_{TA} = table (robot flange) angle.

The Fourier coefficients derived from the Step-Tumble test fit are then used to calculate the coefficients of the gyro performance model according to the following table (34:411). The subscripts "N" and "S" of the Fourier coefficients indicate north and south orientations, and A_L represents local Latitude Angle.

Table 2.4

Relationship of Gyro Drift Coefficients to Fourier Coefficients of OA|EA Tumble Test Data

Gyro Drift Coefficient	Fourier Coefficients
D_F	$\frac{C_{0N} + C_{0S}}{2}$
D_I	$-\frac{C_{1N} + C_{1S}}{2 \cos A_L}$
D_S	$\frac{C_{2S} - C_{2N}}{2 \cos A_L}$
D_O	$\frac{C_{0N} - C_{0S}}{2 \sin A_L}$
D_{IS}	$\frac{2C_{3N}}{\cos^2 A_L}$
D_{II}	$\frac{- C_4}{\cos^2 A_L}$
D_{SS}	$\frac{C_4}{\cos^2 A_L}$
D_{IO}	$\frac{C_{1S} - C_{1N}}{\sin 2A_L}$
D_{OS}	$-\frac{C_{2N} + C_{2S}}{\sin 2A_L}$

For derivation of the above table, see Reference 34, pages 374-412.

Experiment Setup and Methodology.

Hardware. The hardware for this test is summarized in Table 2.4.

Table 2.5

Hardware Used to Conduct Gyro Testing

Gyroscope:	Humphrey Model RG51-0106-1 SDOF Gyro
Robot Arm:	PUMA 560 Unimate Controller Z-100 PC used as a Smart Terminal
Power Supplies:	28 VDC, Sorensen Nobatron DCR60-13 3 VDC, Datel Voltage Calibrator, Model DCV 8500
Other:	Dana Digital Multimeter Model 5900 Cables

The gyro used for the experiment was a Humphrey Model RG51-0106-1, a conventional single-degree-of-freedom (SDOF) torque-rebalanced rate gyroscope. The PUMA 560 Robot Arm was used as the test platform. The gyro was powered by a 28-volt DC power supply. A separate reference voltage of 3 volts was applied. Output from the gyro was directed to the digital multimeter and read and recorded by the test engineer.

<u>Procedure</u>. The gyro was mounted to a metal support base which was in turn attached to the robot flange. The step-tumble test required the following gyro orientations to separate the drift coefficients for the gyro:

(1) Gyro OA parallel to the earth's spin axis (EA) pointing north, IA pointing west at the start of the rotations (OA ∥ + EA)

(2) OA parallel to EA pointing south, IA pointing west at the start of the rotations (OA ∥ - EA)

To align the gyro with the EA it was first necessary to determine the relationship between the PUMA World Coordinate System (WCS) and the EA. To find the WCS relative to EA it was necessary to know the latitude of the robot and the direction of True North with respect to the robot. This information was readily available for the test site and was used to determine the proper robot joint angles to align the gyro OA with the EA.

Once the OA and IA were properly aligned, the gyro was stepped through 360 degrees of rotation by rotating the flange 360 degrees clockwise (cw) followed by 360 degrees counterclockwise (ccw). At each 45-degree increment the robot paused 10 seconds to allow the engineer time to read and record the output. One cw and ccw rotation of the flange for each orientation constituted one set of data for each step-tumble test. Eight sets of data were collected with OA south and eight with OA north (a total of 128 points in each direction).

Software Development. The software was written for the robot's VAL II operating system which was accessed through a Zenith 100 (Z-100) running communication software to act as a smart terminal. The programs, written in the VAL II language, positioned the robot arm for each of the required gyro orientations and rotations. The software listings are in Appendix D.

Results. The statistical package BMDP was used to perform the least squares fit of the output voltage to the model of Equation 2.28. The Fourier coefficients obtained from this fit are listed in Table D.1 of Appendix D. Table 2.5 summarizes the drift coefficients (and their standard error) of the performance model equation.

Table 2.6

Performance Model Equation Coefficients

Drift Coefficient	Calculated Value	Standard Error
D_F	1.49999	0.00188
D_I	0.00249	0.00031
D_S	0.07619	0.00031
D_O	0.00188	0.00295
D_{IS}	0.00117	0.00035
D_{II}	0.00107	0.00035
D_{SS}	0.00107	0.00035
D_{OI}	0.00389	0.00036
D_{OS}	0.00120	0.00036

Since the duration of the tests was approximately three hours and the gyro's output axis was aligned with the earth's rotational axis, error sources did not include earth rate. All drift coefficients except D_O were significant. From previous rate-table tests D_F was determined to be 1.5 volts. Except for D_F, there was no test data with which to compare the drift coefficients. However, the coefficients are reasonable, and as with the accelerometer fourpoint tests indicated the feasibility of using the robot arm for testing inertial sensors.

The main purpose of the gyro test was to demonstrate the robot arm's ease of reconfigurability and its maneuverability and therefore its usefulness as a multi-purpose testbed. This was clearly demonstrated by the gyro step-tumble tests.

Robot Performance Criteria

A final important aspect of determining technical feasibility is to answer the questions:

(1) What are the robot performance criteria for inertial sensor/system testing?

(2) Which of today's robots meet those criteria?

Determination of Performance Criteria. All the criteria for selecting a robot for industrial applications are fully described in the robotics literature (10:214-301; 17:263-272; 22). In this thesis we will address only the criteria pertinent to inertial sensor/system testing. The term "system" here refers to the inertial measurement unit (IMU), not a complete inertial guidance system.

Load Requirement. Today's inertial sensors weigh five to ten pounds, and IMU's weigh up to twenty-five pounds (8). The weight of the sensor/system plus its mount or gripper determined the robot load requirement for this research.

Drive Method and Number of Axes. The class of robots considered was electric motor driven manipulators with six degrees of freedom. Electric motor drives were selected because they are the most accurate over the load requirement range (10:109-111). While six degrees of freedom reduces stiffness with some resultant loss of accuracy, it allows a greater variety of configurations. This makes the robot adaptable to a wider range of test situations and allows for the development of new types of tests not possible on existing test equipment.

Axis Rotation. The static gyro and accelerometer tests demonstrated the need for at least one joint to rotate 360 degrees or more. This capability is most frequently available at the robot wrist (the PUMA 560 flange, for example, was a part of the wrist assembly). Other joint rotations are important also, as in the gyro step-tumble test. Thus wrist pitch, roll, or yaw of 360 degrees or more is required, and maximum rotation of the other three joints should be at least 180 degrees.

Robot Task Programming. Robot tasks in industrial applications are frequently programmed by manual or lead-through teaching methods. With these methods the robot or a robot simulator is manually directed to each point of the task and the location of the points recorded to build the task program (17:210-216, 266-271). However, this is not practical for the inertial sensors/systems application. The robot must respond more to algorithmic commands than to the repetition of specified point locations, especially in dynamic tests. Off-line programming using either robot programming languages or standard languages provides instructions to move the manipulator, read sensors, send output signals, and many other instructions essential to sensor/system testing (17: 216). Off-line programming also permits the development of several programs at once. Therefore off-line programming capability is essential.

Positioning Accuracy. Positioning accuracy is another characteristic to consider in robot selection. Positioning accuracy is defined as "the difference between the position desired and the position actually achieved" (10:76). Repeatability is a statistical term associated with accuracy. It is a measure of the difference between successive movements to the same commanded position (10:76). Since the accuracy depends upon the particular load that the gripper carries, most robot manufacturers provide a numerical value for repeatability rather than accuracy (17:19). The requirement for inertial sensor/system testing is to achieve the greatest positioning accuracy and repeatability possible. For today's robots that means a repeatability of 0.010 inches or less. (Accuracy issues are discussed further in Chapter IV.)

Other Criteria. Variable acceleration and/or deceleration capability is an asset, and in fact a requirement for some dynamic testing.

The robot mount is a final consideration. Robot manipulators may be mounted to the floor, the wall, or overhead (gantry). In general the author believes that the floor mount is most desirable. Floor "mounted" robots can be moved from one location to another fairly easily for the situation in which they are being used as a "quick-look" test stand at different test stations. A floor mounted robot is more stable and less susceptible to positioning errors caused by joint and link flexures than a wall mount, and does not require the elaborate installation structure of a gantry model.

Identification of Suitable Robots. A listing of prospective robots containing their physical characteristics and estimated base prices was obtained (29) using a commercial computer package called "Robot Search Program" (Robot Analysis Associates, Inc.). Because this listing was compiled in the early stages of determining robot requirements for inertial sensors testing, the criteria used were more general than those of the discussion above.

Therefore, the first step in identifying those robots suitable for inertial testing was to enter their performance characteristics in an electronic spreadsheet (Lotus 1-2-3). Repeated application of the "Sort" and "Query" functions of Lotus were then used to highlight the manipulators with the maximum performance capabilities (9:435-448). This resulted in a field that met the criteria listed in the previous section. (An excellent summary of spreadsheets and discussion of their uses in other aspects of engineering can be found in Reference 20.)

The field of prospective robots was finally narrowed to the following for purposes of comparison with non-robotic units (8;2) and economic analysis:

Name	Mount		Max Rotation (Wrist)	Other Rot	Joint	Max Load (lbs)	Accuracy (ins)	at (ips)	Variable Accel/Dec
A'matix AID-900	Floor	R	440	P315	Wrist	66	0.008	30	Y
Yaskawa V-12	F/O/W	P	360	YR330	Wrist	26	0.008	80	Y
Cinn Mil T3-646	Floor	R	900	PY238	Wrist	50	0.010	25	N/A
PUMA 560	Floor	R	532	P200	Wrist	5.5	0.004	20	N

Name	Mount	Max Rotation	Other Rot	Joint	Max Load (lbs)	Accuracy (arcsec per axis)	at (ips)	Variable Accel/D
Vertical Table	Floor	Continuous			50	(1		N
2-axis Contraves	Floor	Continuous			75	1		N
3-axis Contraves	Floor	Continuous			100	*3		N

*Difference in accuracy due to different type of bearings, not number of axes

**Estimated cost of new 3-axis table

Figure 2.4. Spreadsheet of Robotic and Non-Robotic Testing Units

The final choices from the spreadsheet analysis are the first three on the list. The PUMA 560 is included because it was the robot used in this research; it was not selected by the Robot Search Program because of its maximum load of only 5.5 pounds.

The non-robotic tables have the advantage of continuous rotation and accuracies in the arcseconds range. However, the load capabilities are comparable, including the 100-pound load. For example, in addition to the robots listed above, the Cincinnati Milacron T3-776 meets the rotational and accuracy requirements while carrying a load of 150 pounds. The robotic testbeds, however, are more versatile and less expensive and have other potentials which are discussed in Chapter 5.

Summary

In this chapter we have investigated the technical feasibility of using robots as testbeds for inertial sensors. The accelerometer and gyro tests showed the versatility of a robot arm and that the arm offers sufficient precision for at least quick-look tests and quite possibly for precision characterization tests. Present-day industrial robots were located which meet the performance criteria for inertial sensor testing. In summary, robots are technically feasible testbeds for inertial sensors (or IMU's). In the next chapter the issue of economic feasibility is addressed.

III. Economic Feasibility

Once technical feasibility has been established, the next important question is, "Is the proposal economically feasible?" This chapter assesses economic feasibility by performing a life cycle costing analysis for both the robotic and non-robotic testing units.

Economic Analysis Criteria

A vertical table, and 2-axis Contraves table, and a 3-axis Contraves table are the non-robotic testing units. Table 3.1 shows the performance selection criteria of the four robots, the performance characteristics of the testing tables, and base prices for all units.

Table 3.1

Performance Characteristics and Base Prices of Robotic and Non-robotic Testing Units

Name	Mount	Max Rotation (Wrist)	Other Rot	Joint	Max Load (lbs)	Accuracy (ins)	at (ips)	Variable Accel/Dec	Base Price
A'matix AID-900	Floor	R 440	P315	Wrist	66	0.008	30	Y	50000
Yaskawa V-12	F/O/W	P 360	YR330	Wrist	26	0.008	80	Y	69600
Cinn Mil T3-646	Floor	R 900	PY238	Wrist	50	0.010	25	N/A	70000
PUMA 560	Floor	R 532	P200	Wrist	5.5	0.004	20	N	80000

Name	Mount	Max Rotation	Other Rot	Joint	Max Load (lbs)	Accuracy (arcsec per axis)	at (ips)	Variable Accel/Dec	Base Price
Vertical Table	Floor	Continuous			50	<1		N	150000
2-axis Contraves	Floor	Continuous			75	1		N	500000
3-axis Contraves	Floor	Continuous			100	*3		N	** 3000000

*Difference in accuracy due to different type of bearings, not number of axes

**Estimated cost of new 3-axis table

Estimated robot prices, even base prices, varied widely. Test table costs were obtained from the Central Inertial Guidance Test Facility (CIGTF), Holloman Air Force Base, New Mexico (8;2).

Life Cycle Costing

Life Span Costs. Life Cycle Costing (LCC) defines "life span" as the time to replacement or major modification (mechanical or electronic). Even though major mechanical modifications may occur only every ten years or so, advances in electronics determine a life span of three to five years for robotic and non-robotic testing units alike (1:20; 2). Therefore, a life span of five years was chosen for the analysis.

Once the life span is determined, three specific costs must be analyzed (15:66-67):

(1) Research and developments costs

(2) Investment costs

(3) Operational costs.

The variation of the costs must be estimated for each year of the unit's life. The costs are totaled for each year and then summed to calculate the LCC via the equation:

$$LCC = \sum_{i=1}^{N} \frac{R_i}{(1+r)^i} \qquad (3.1)$$

where LCC = Life Cycle Cost
R_i = Yearly Costs (R & D + Investment + Operational)
i = Year number
r = Interest rate (10% assumed).

Lotus 1-2-3 automatically calculated the LCC. Formulas were entered into the spreadsheet cells, which facilitated rapid and efficient economic analysis without writing any computer programs. More detailed LCC determination can include calculating cost ranges for each item, assigning probabilities to the estimates, and thus including random variables, which does require some programming. The results can then be compared via a spreadsheet analysis.

Research and Development Costs. Research and development (R & D) costs are defined as "the resources required to develop the new capa bility to the point where it can be introduced into the operational inventory at some desired level of reliability" (15:66). The economic study is for the development of a prototype robotic test station, and thus the R & D costs are greater than they would be for a previously developed robotic test station. R & D estimates included in the study were:

- Investigation of which "standard" tests can be performed on a robot arm
- Research to determine which robot best suits the application
- Development of software for tests
- Installation of precision measuring device, such as lasers (or adaptation of lasers presently used in robotics (19))
- Development of new "non-standard" tests using the full robotics capabilities:
    (1) Theory
    (2) Software
    (3) Testing and comparison with results of standard tests.

Based on the equivalence of one man-year of work plus computer time, the first year's R & D cost was estimated to be $40,000, 15% for year 2, 5% for year 3, and 1% for each of years 4 and 5. The continuing R & D costs were for ongoing development of new tests and study of the applicability of new robots on the market.

R & D costs for the vertical and 2-axis tables were 15% of the robot costs for the first year, to account for changes in electronics which could be introduced. However, the new 3-axis table being developed was treated the same as a prototype robot for the first 3 years, using the same R & D percentages.

Investment Costs. Investment costs are one-time outlays required to introduce some capability into the operational inventory (15:66). Investment estimates were adapted from manufacturing applications (1) as follow:

1. Base price. For this study the base price included the robot arm, controller, and teach pendant.
2. Support costs--15% of the base price.
   - Additional disks
   - Interface with existing data acquisition hardware and software
   - Installation
     (1) Mounts
     (2) Safety arrangements
3. "Soft" costs--25% of the base price.
   - Training of personnel
     (1) Hardware

(2) Software

- Programming/Systems analysis
- Integrated data acquisition system documentation

Table 3.2

Examples of Life Cycle Costing Worksheet

Cinn Mil T3-646	Year 1	Year 2	Year 3	Year 4	Year 5	
R&D	40000	6000	2000	2000	2000	
Investment	0	98000	9800	0	0	
Operating	0	19600	19600	9800	24500	
Totals	40000	123600	31400	11800	26500	
LCC	36364	102149	23591	8060	16454	186618

Vertical Table	Year 1	Year 2	Year 3	Year 4	Year 5	
R&D	6000	0	0	0	0	
Investment	0	177000	17700	0	0	
Operating	0	17000	10000	10000	17000	
Totals	6000	194000	27700	10000	17000	
LCC	5455	160331	20811	6830	10556	203982

Investment costs were spread over a two-year period, with 100% of the Initial Total Cost (Table 3.2) in year 2 and an additional 10% for unaccounted-for costs in year 3. Investment cost allocation was the same for robots and testing tables.

<u>Operating Costs</u>. Operating costs, the "recurring outlays required year by year to operate and maintain the capability in service over a period of years," (15:67) included:

- Maintenance

(1) Parts replacement

(2) General upkeep of arm (periodic oiling of joints, calibration, etc.)

- Periodic personnel training.

For the robots, years 2 and 3 were estimated at 20% of the total investment cost, year 4 was 10% of the total, and year 5 included periodic retraining costs and was estimated at 25% of the total investment cost.

More specific figures are available for the testing tables (2). Operating costs for years 2 and 5 included both maintenance and personnel training; years 3 and 4 were maintenance expense only. Table 3.3 is an example of the LCC, as calculated and displayed in the spreadsheet.

The life cycle costs of the robot arms and test tables were:

Table 3.3

Total Life Cycle Costs

Device	LCC
Automatix AID-900	$ 146,279
Cincinnati Milacron T3-646	186,618
Yaskawa V-12	185,811
PUMA 560	206,787
Vertical Table	203,982
2-axis Contraves	522,239
3-axis Contraves	2,818,062.

Results

From the above analysis it is feasible that a prototype robotic test station, the T3-646 for instance, could replace one table, perhaps the vertical table, with a resultant decrease in LCC of $17,364. Of course the savings increase substantially if the robot replaces the 2- or 3-axis tables.

Another important advantage and source of savings is the versatility of a robot arm. Over the long term both standard and experimental inertial instrument tests can be performed by simply re-programming the robot, rather than rebuilding or developing a new test table. In the short term, as was the case for the gyro tests, the robot can be quickly reconfigured at any point in the test with no manual readjustments involved.

The groundwork for a prototype effort by a testing organization exists and is ready to be applied.

IV. Limitations

In this chapter the technical limitations of robots are discussed in detail as they relate to testing. Practical engineering limits, computer modeling limits, and measurement and instrumentation limits are examined and related to the sensor testbed application.

Practical Engineering Limits

Practical engineering limits include flexure of the links and joints of the robot arm, which will vary from one robot to another. A simple test using the PUMA 560 arm and the Robotic Simulation (ROBSIM) program demonstrates this limitation. Robot control schemes can also limit the accuracy of trajectory tracking.

Flexibility of the Robot Arm. The PUMA 560 has six joints. This experiment uses only two of these joints, the shoulder and the flange. As the shoulder link rotates from the vertical, it exerts a large moment about the base y-axis due to gravity (see Figure 2.1). This produces a bend in the robot arm which should be measurable by a high accuracy accelerometer mounted on the end of the arm. The flange on the other hand, with a smaller radius of rotation and mass, exerts a much smaller moment about its x-axis.

The Systron-Donner 4841F accelerometer was first mounted on the flange, then the flange was rotated from the vertical 90° and back about its x-axis in 10° increments. The experiment was then repeated in the same configuration but with the flange fixed and the shoulder rotated in

10° increments about the base y-axis starting from a vertical position. These results are shown in Table 4.1.

Table 4.1. Rotation Alignment Errors

Degrees From Vertical		Shoulder Error (g)	Flange Error (g)
0	Minimum	0.000	0.000
10		0.006	0.006
20		0.012	0.012
30		0.019	0.018
40		0.025	0.023
50		0.029	0.028
60		0.034	0.031
70		0.036	0.034
80		0.038	0.035
90	Maximum	0.039	0.036
80		0.037	0.035
70		0.036	0.034
60		0.034	0.032
50		0.029	0.028
40		0.025	0.024
30		0.019	0.019
20		0.012	0.012
10		0.007	0.006
0	Minimum	0.000	0.000

Figure 4.1 is a plot of the actuator torque versus time for the shoulder rotation. The plot was generated by the Robotic Simulation (ROBSIM) discussed in the "Computer Modeling Limitations" section which follows. ROBSIM is based on a rigid-link mechanical model (14:2).

Table 4.1 demonstrates the inaccuracies of robot positioning and indicates that the flexibility of the robot arm should be a consideration when precise positioning and orientation is needed. Figure 4.1 shows that the torque is a function of the robot orientation and that the orientation errors are due in part to mechanical flexure.

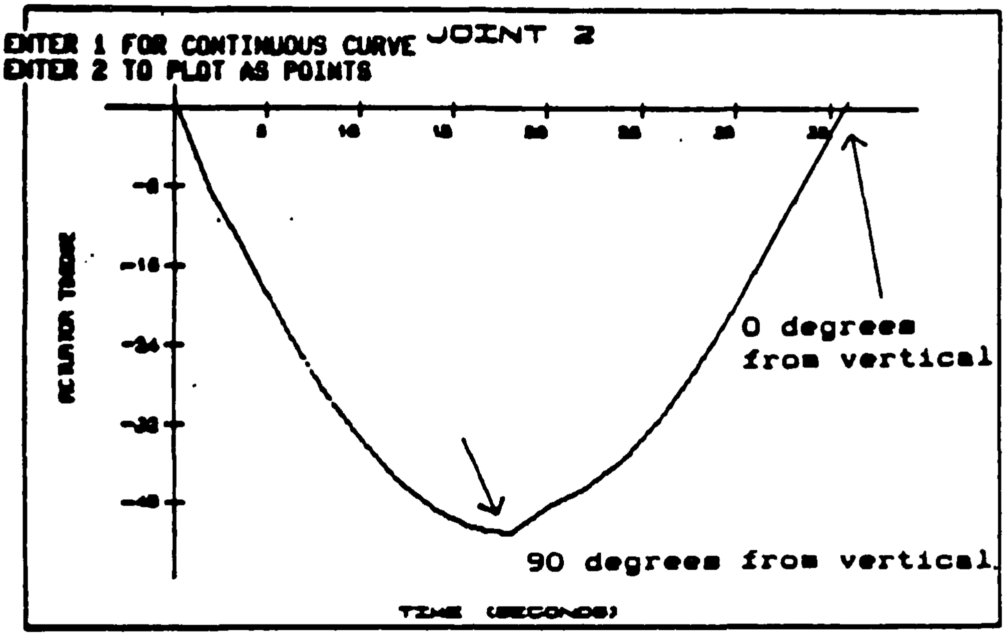


Figure 4.1. ROBSIM Output for the PUMA 560 Shoulder Rotation

Control Schemes. Robot control is limited by control method and unmodelled forces, and by the restrictions of robotic programming languages.

The most widely used control method today applies a separate axial control loop for each joint designed with linear-control laws (17:80), often with fixed gain (17:72). The required gain is highly dependent on the moment of inertia at each joint of the robot arm which in turn varies with the arm position and robot payload. A variety of schemes, including adaptive control, have been proposed and implemented (17:51-

81), but research is still being done to represent previously unmodelled forces (18) and implement adaptive control.

Robotic programming languages, too, can be a control limitation in that they often do not include the facilities to implement complex mathematical formulas. One must bypass the robot operating system to implement experimental control techniques and gain greater precision.

Computer Modeling Limits

An important element in the effective use of robots and in designing unique inertial sensor tests for a robot is an accurate and comprehensive computer simulation program. Simulation programs are being developed in several different environments, including universities such as Arizona State University (24) and AFIT (6) as well as private industry. The Robotic Simulation (ROBSIM) program was installed and studied to determine both its advantages and its shortcomings. A brief overview of ROBSIM's capabilities is included in Appendix E.

Applications. In many cases, computer simulation is directed strictly to industrial applications, including multiple arms, creation of workstations and assembly lines, and so forth. ROBSIM includes the industrial applications, but it also allows for simulation of different types of control schemes and for the creation of data files and plots of the robot forces, torques, etc. for analysis purposes. This makes it attractive for the testbed application.

Mechanical Model. The kinematic (and dynamic) analysis tools implemented in ROBSIM are based on a rigid-link model of serial, open-loop kinematic chains with one-degree-of-freedom joints. (Details and examples may be found in the "Kinematics Analysis" section of Reference

13.) Although this is not completely realistic, it does provide help in determining the behavior of the robot and the identification of possible flexure errors, as demonstrated in the flexure test of the previous section. The comprehensiveness and accuracy of the simulator's mechanical model of a specific arm is critical in determining the validity of the simulation.

Dynamics Model. An accurate dynamics model is also essential for acceptable simulation. ROBSIM uses homogeneous transformation matrices for calculating transformations between arbitrary sets of coordinates.

The difficulty with robotics dynamics models relative to inertial navigation models lies in the differences of notation. There has as yet been little cohesion established between classical methods of inertial navigation and methods of describing the dynamics of robots. A means of melding the two fields is needed. The beginning of that melding is to parallel the dynamics equations and error analysis methods developed for robotics with the classical inertial navigation techniques. A notation to accomplish cohesiveness is developed and presented in Appendix F, followed by a tutorial presentation of introductory robot kinematics using inertial navigation notation.

Measurement and Instrumentation Limits

The tests and performance criteria study of Chapter II demonstrated the positioning and measurement accuracy limitations of present-day robots. Data acquisition ability is restricted by the fact that robot controllers do not usually include high-accuracy analog-to-digital converters and the rapid sampling ability necessary for sensor testing (although either 12- or 16-bit converters can be installed). Possible solutions to the measurement and instrumentation limits are discussed in Chapter V.

Summary

In this chapter the robot and robotic simulation limitations which affect the inertial sensor testing application have been addressed. The final chapter will summarize both the robot's advantages and its shortcomings and will discuss robotic potentials.

V. Conclusions and Recommendations

Conclusions

This research included theory and applications of robots, sensors, and robot computer simulations. Determination of robot performance criteria, suitable robots, and economic analysis of robotic and non-robotic test units were accomplished. Robotic limitations were also studied. All of these areas show that there is potential in applying robotics to inertial sensor testing.

In an attempt to control robots more precisely and to interface with computers (and computer simulations) other than the robot's particular controller, research is in progress to control robots from computers such as the VAX 11-780 (AFIT, NASA Langley) or interface with such computers for control and data acquisition (for example, Cincinnati Milacron's Robot Offline Programming System, or ROPS).

From the study presented here, robots large and small could begin to be used as checkout testbeds for inertial sensors, possibly in such applications as immediate flightline checkout of sensors or IMU's suspected of being inoperable rather than sending them away to a depot for checkout.

Robots can be multi-purpose testbeds for performing standard tests on inertial sensors, and the potential for devising unique inertial sensor/system tests exists. Robots with variable acceleration/deceleration and a large rotational range suggest dynamic test possibilities that have not yet been explored. Perhaps a subjecting the sensor/system to a helical motion, or to a rapid swinging motion of the robot followed by a sudden deceleration would excite sensor/system error terms without to enhance or replace centrifuge or other testing. Variations of system trajectories could be tracked with lasers and the system errors analyzed by comparison with the laser data. With extensive computer simulation capabilities such as those of ROBSIM, engineering theory could devise new tests which would be efficiently and safely produced on the simulator, saving both time and money. The simulator-robot combination would encourage engineering creativity, an important commodity in the realm of research and development, where new tests and testing units are needed to keep pace with hardware developments (2).

Recommendations

This research raises further questions. Are robots feasible for system tests? Can the limitations be overcome? What should be done to extend the work presented here?

The solution for robot accuracy constraints may lie not in improving the robot's precision, but rather in providing precision reference measurements for use in sensor output analysis. Laser technology and other instrumentation advances have the potential to accomplish this. For example, providing precision through reference measurement is already in use in noisy, imprecise environments such as the test track at Holloman Air Force Base; and laser technology is currently being used for robot positioning accuracy (19). A cost analysis for laser or other precision measurement technology should be accomplished to extend the economic feasibility study.

The potential for testing precision senors/systems should be further determined by noise characterization of the robot arm. In addition, the sensors used in this study, or similar sensors, should be tested under more controlled laboratory conditions and compared to test results from non-robotic units.

It is also recommended that test engineers and analysts take a new look at the possibilities for dynamic tests using robotic capabilities and begin devising those tests. The groundwork for a prototype effort has been presented in this study and is recommended for future implementation.

Appendix A. Robot Safety Summary

General Procedures

Certain safety issues are important enough to be included in the list of warnings in this appendix (30). These cautions should be heeded by anyone intending to operate the robot arm. It is by no means an exhaustive list, nor is it intended to replace the safety summaries in the manuals.

Should the robot behave erratically, or if an emergency arises, the operator must stop the robot motion (see procedure below), turn ARM POWER OFF, and report the problem to an appropriate source.

Specific Precautions

Before Turning the Arm Power On.

DO NOT replace components or make adjustments to the equipment with the electrical supply turned on.

DO ensure that all personnel are clear of the robot operation area. Should a malfunction occur, robot arm behavior will be unpredictable.

DO ensure that the working area is kept clear of all obstacles.

DO set the velocity at a slow speed (SP 10 for the PUMA), especially when testing a new program. This allows more time to react should the program work incorrectly. Once a program works properly, the speed may be increased.

DO ensure that at least one other authorized person is present at all times.

While Robot Power is On.

DO NOT operate the robot in the FREE mode.

DO keep one hand near the ARM POWER OFF "panic button" located on the teach pendant at all times. This allows for emergency shutdown when necessary.

DO NOT shut power off while the robot arm is moving except for an emergency stop.

DO NOT try to stop program execution with a CONTROL-C command. It will not work.

When Interfacing to the Robot and Controller.

DO NOT connect or disconnect cables to any connection port unless power has been removed from the robot arm.

DO ensure that the length of bolts used to attach components to the wrist mounting flange do not extend beyond the hand mounting surface by more than 1/4 inch. Longer bolts interfere with and keep the wrist from turning, which will cause damage to the wrist.

Emergency Shutdown

There are a number of ways to shut down robot power while a program is running. If an emergency arises, any of the following actions will suspend operation:

Pressing the ARM POWER OFF button on either the controller or the teach pendant will cause an immediate stop of robot motion. This should not be done routinely.

Entering the PANIC command accomplishes the exact same thing.

Turning the RESTART/HALT/RUN switch on the controller to HALT will also stop motion immediately. Turning it back to RUN will resume program execution.

Entering the ABORT command will first complete the current program step before stopping robot motion. (PROCEED will resume program execution.) Routine halts should use this command.

Appendix B. Software for Vertical-Seeking Test

Overview

The software was designed to implement the theory expressed in the main text. The accelerometer was interfaced to the VAL II operating system using an Input/Output Module. The VAL II operating system reads the accelerometer output during the rotation of a joint and stops the corresponding joint at the point of recorded the lowest accelerometer reading. The software is implemented using a main program (GRAVITY.MAIN) and two calling subroutines (GRAVITY.MAX and DELAY).

Program Descriptions

GRAVITY.MAIN. The GRAVITY.MAIN program positions the arm before calls to the GRAVITY.MAX subroutine. The PUMA 560 is placed in the ready position (Figure B.1) where the axis of rotation of Joint 5 is parallel to the y-axis. The GRAVITY.MAX subroutine is called using Joint 5, and the position of the PUMA 560 robot is recorded. Note that Joint 5 is rotated -90 degrees before rotating because the input axis of the accelerometer is mounted at a 90-degree angle to the tool z-axis. The PUMA 560 base is then rotated 90 degrees, thereby placing the axis of rotation of Joint 5 parallel to the x-axis. The GRAVITY.MAX subroutine is again called using Joint 5, and the position of the PUMA 560 robot is recorded. Two vectors are formed from the two points recorded and their cross-product is calculated. The PUMA 560 is then positioned to align its tool z-axis with this vector.

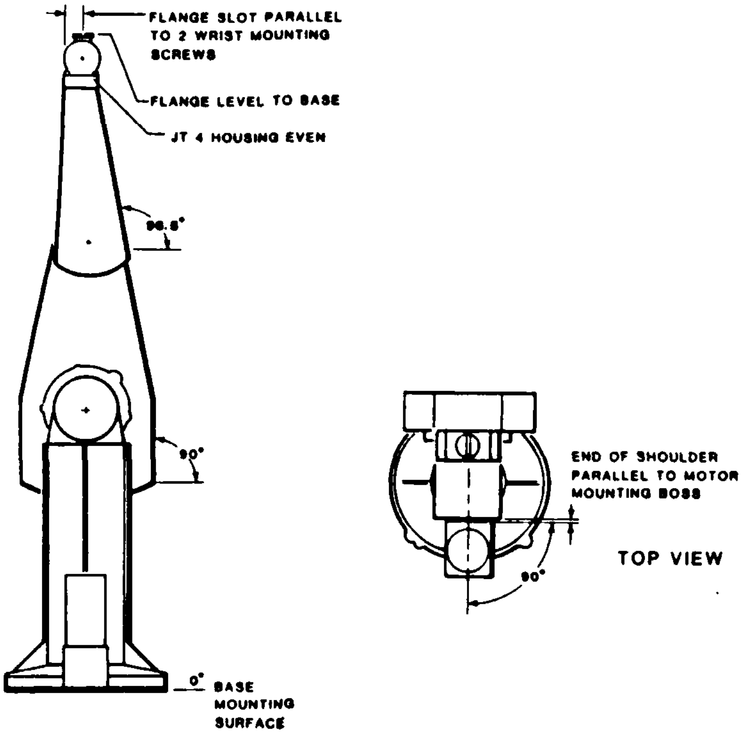

FLANGE SLOT PARALLEL TO 2 WRIST MOUNTING SCREWS
FLANGE LEVEL TO BASE
JT 4 HOUSING EVEN
96.5°
90°
0°
BASE MOUNTING SURFACE
END OF SHOULDER PARALLEL TO MOTOR MOUNTING BOSS
TOP VIEW
90°

Figure B.1. READY Position of PUMA 560
(Reference 30)

<u>GRAVITY.MAX</u>. The subroutine GRAVITY.MAX determines the minimum accelerometer reading using an iterative process. The "DRIVE" command allows a single joint to be rotated through a specified angle. The accelerometer readings are compared at the beginning and end points of each "DRIVE" command. If the absolute value of the accelerometer reading is larger at the end point, the arm will continue to move in the same direction. If not, the arm will reverse the direction of movement and decrease the angle to be moved through, thereby allowing the robot arm to stop at points between the last beginning and end points. The loop is broken out of if the move angle is less than 0.005 degrees (the

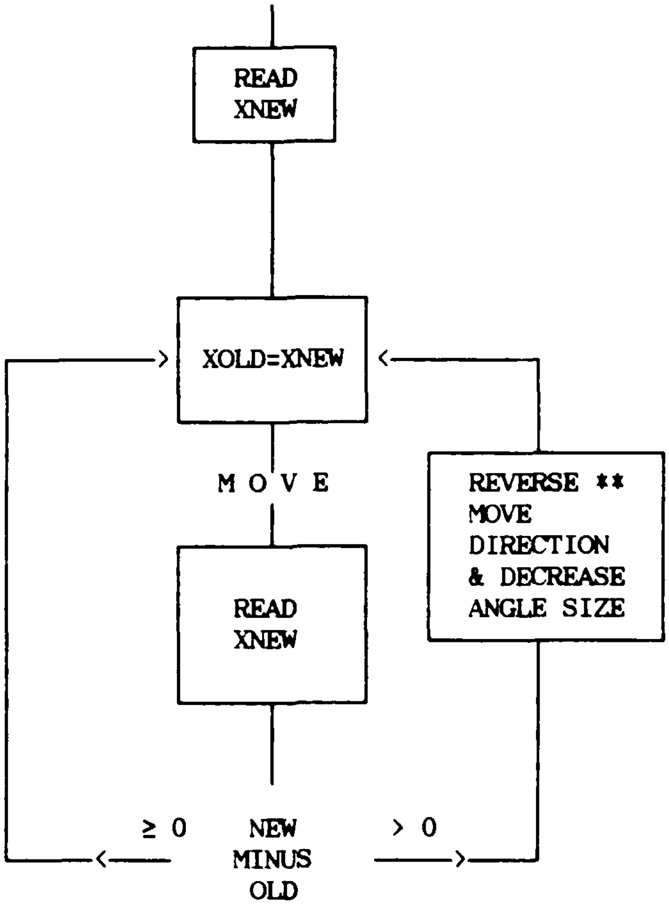
limit of movement definition in the robot arm), or if the accelerometer reading is zero. A flow chart is shown on the following page.

DELAY. The DELAY program is a useful subroutine which causes program execution to pause for a specified amount of time. To use, simply define the variable DTIME in seconds (integers only) and then CALL DELAY. The program will pause for DTIME seconds. Examples of use are in all programs in this appendix. The delay is only an approximation and should not be used for precision timing.

The principle behind the DELAY subroutine is that the WAIT command without additional parameters (30) will cause a pause of 28 milliseconds. The WAIT command is put in a loop which repeats 37 times for every delay of one second. It is assumed that other commands in the loop take relatively little time compared to the WAIT command. The DELAY subroutine DOES NOT cause the motion of the arm to stop. The user is strongly reminded that after a robot motion command is given, the program will continue to execute. The DELAY program was used together with the BREAK command. The BREAK command suspends program execution until previous arm motion is completed. The DELAY subroutine is used simply to allow the accelerometer readings to stabilize and allow time for recording of data.

FOR THE LISTINGS OF THE VAL II PROGRAMS (pages 54-55), CONTACT:

Dr. Peter Maybeck
Air Force Institute of Technology
School of Engineering
Wright-Patterson AFB, Ohio 45433-6583



** When the angle size < .005 degrees, the subroutine will return to the main program.

Figure B.2. Flowchart for Subroutine GRAVITY.MAX

Appendix C. Accelerometer Fourpoint Test

Theory

The performance model equation for an accelerometer is (33:A-1):

$$A_{ind} = E/K_1 = K_0 + a_i + K_2 a_i^2 + K_3 a_i^3 + \delta_o a_p - \delta_p a_o$$
$$+ K_{ip} a_i a_p + K_{io} a_i a_o + K_{op} a_o a_p + K_{pp} a_p^2$$
$$+ K_{oo} a_o^2 + K_{ppp} a_p^3 + K_{ooo} a_o^3 \qquad (C.1)$$

where

a_i = Acceleration component along input axis (g)

a_p = Acceleration component along pendulous axis (g)

a_o = Acceleration component along output axis (g)

E = Accelerometer output

A_{ind} = Indicated acceleration (g)

K_0 = Bias (g)

K_1 = Scale factor (output units/g)

K_2 = Second order non-linearity coefficient (g/g^2)

K_3 = Third order non-linearity coefficient (g/g^2)

δ_o = Input axis misalignment about output axis (radians)

δ_p = Input axis misalignment about pendulous axis (radians)

K_{ip} = Cross coupling coefficient (g/g^2)

K_{io} = Cross coupling coefficient (g/g^2)

K_{op} = Cross coupling coefficient (g/g^2)

K_{oo} = Second-order cross axis coefficient (g/g^2)

K_{pp} = Second-order cross axis coefficient (g/g^2)

K_{ooo} = Third-order cross axis coefficient (g/g^3)

K_{ppp} = Third-order cross axis coefficient (g/g^3).

However, when an accelerometer is rotated about its output axis, the accelerations about that axis become zero, reducing the model to (32:C-4):

$$A_{ind} = E/K_1 = K_0 + a_i + K_2 a_i^2 + K_3 a_i^3 + \delta_o a_p + K_{pp} a_p^2 + K_{ppp} a_p^3 . \quad (C.2)$$

The performance-model equation is further reduced at the four positions of 90°, 270°, 0°, and 180° as follows (32:C-4):

a. At the 90° position, $a_i = +1g$ and $a_p = 0g$, so

$$E(90^\circ) = (K_0 + 1 + K_2 + K_3)K_1 . \quad (C.3)$$

b. At the 270° position, $a_i = -1g$ and $a_p = 0g$, so

$$E(270^\circ) = (K_0 - 1 + K_2 - K_3)K_1 . \quad (C.4)$$

c. At the 0° position, $a_i = 0g$ and $a_p = +1g$, so

$$E(0^\circ) = (K_0 + \delta_o + K_{pp} + K_{ppp})K_1 . \quad (C.5)$$

d. At the 180° position, $a_i = 0g$ and $a_p = -1g$, so

$$E(180^\circ) = (K_0 - \delta_o + K_{pp} - K_{ppp})K_1 . \quad (C.6)$$

These are the equations, then, that are used to calculate the error coefficients from the fourpoint tests.

Software

Following are the listings of the VAL II programs DELAY and FOUR used to perform the accelerometer fourpoint test.

FOR THE LISTINGS OF DELAY AND FOUR (pages 58 and 59), CONTACT:

Dr. Peter Maybeck
Air Force Institute of Technology
School of Engineering
Wright-Patterson AFB, Ohio 45433-6583

The following section is a listing of the terminal display when executing the program FOUR. Accelerometer outputs are sent directly to the voltmeter rather than to the UNIMATE controller's analog-to-digital converter.

```
.cal
Are you sure (Y/N)? y
.do align
.do ready
.ex four
***PLACE THE ARM IN THE 'READY' POSITION AND
   ENTER 'PROCEED' TO CONTINUE
(PAUSED)
Stopped at four, step 18
.proceed

HOW MUCH TIME TO READ EACH OUTPUT (SEC)?15
NUMBER OF FOURPOINTS TO RUN?20

BEGINNING FOURPOINT NO.  1.
COMPLETION OF FOURPOINT NO.  1.
BEGINNING FOURPOINT NO.  2.
COMPLETION OF FOURPOINT NO.  2.
           .
           .
           .
BEGINNING FOURPOINT NO.  20.
COMPLETION OF FOURPOINT NO.  20.
COMPLETION OF FOURPOINT TESTS
Program completed
Stopped at four, step 68
```

A listing of the joint coordinates recorded at each rotation can be obtained by the "llist", or location list command. A sample of the location list for the fourpoint test follows.

```
.llist

              X/Jt1    Y/Jt2    Z/Jt3    O/Jt4    A/Jt5    T/Jt6
 neg.g[1]    -20.38   205.34   864.94  179.995   -0.066  179.995
 pos.g[1]    -20.38   205.34   864.94  179.995   -0.066   -0.016
 start         0.00     0.00     0.00   90.000  -90.000    0.000
 start.horizontal    -20.44   149.09   921.13   90.000  -89.923    0.000
 start.vertical    -20.38   205.34   864.94  179.995   -0.066   -0.016
 zerog.0[1]    -20.38   205.34   864.94  179.995   -0.066  -90.006
 zerog.180[1]    -20.38   205.34   864.94  179.995   -0.066   89.984
```

Appendix D. VAL II Gyro Programs and Output Listings

The VAL II language programs for performing the Step Tumble Tests were:

scalfact.test	
sttest.cp	sttest2.sp
sttest1.sp	delay.sp

The Step Tumble Tests were performed by executing sttest.cp. This executed the two tests eight times each, alternating north and south orientations.

Although the gyro output for this thesis was directed to a digital multimeter, the output can be directed instead to an analog-to-digital converter in the Unimate controller. The converter that was available for this thesis work was only 12 bits and therefore not accurate enough for this test. However, the software was written to accommodate the converter and several orientations were performed to demonstrate how this can be accomplished. To output to the multimeter, simply comment out the commands to the converter.

A list of the VAL II language programs follow. Each one is fully descriptive.

FOR THE LISTINGS OF THE VAL II PROGRAMS (pages 62-65), CONTACT:

Dr. Peter Maybeck
Air Force Institute of Technology
School of Engineering
Wright-Patterson AFB, Ohio 45433-6583

Sample Output Listings Using A to D Converter

```
Performing Step-Tumble test-second orientation.

Perform test with output axis parallel to earth
spin axis and directed north.

DATA RESULTS ARE AS FOLLOWS:

                CLOCKWISE       COUNTERCLOCKWISE

0 DEG                                1.497313
315 DEG         1.497313             1.497313
270 DEG         1.497313             1.489985
225 DEG         1.492428             1.489985
180 DEG         1.497313             1.494871
135 DEG         1.497313             1.497313
90 DEG          1.497313             1.497313
45 DEG          1.504641             1.502198
0 DEG           1.497313

This orientation is done.
Program completed
Stopped at sttest.cp, step 9
```

Table D.1

Fourier Coefficients from
Step-Tumble Test

	North Orientation		South Orientation	
Fourier Coefficient	Calculated Value	Standard Error	Calculated Value	Standard Error
C_0	1.50119	0.00350	1.49878	0.00140
C_1	-0.00305	0.00031	0.00078	0.00017
C_2	-0.00117	0.00031	0.00001	0.00017
C_3	0.00103	0.00031	0.00033	0.00017
C_4	-0.00094	0.00031	0.00016	0.00017

Appendix E. Overview of ROBSIM

Scientific investigation involves the synthesis of mathematical models using intuitive assumptions and insights based on observation. These models are then subject to test and evaluation. The ROBSIM simulation package follows this thinking process by dividing it into three software modules: model synthesis (INITDRVR), simulation (SIMDRVR), and analysis (POSTDRVR). Extensive graphics capabilities accompany all phases of ROBSIM. Although the program is still being modified at NASA and updated in cooperation with AFIT, it can be used effectively to demonstrate robotics simulation with interactive computer graphics.

Synthesis (INITDRVR)

The major advantage of any simulation program is that the system does not have to physically exist before meaningful research can begin. ROBSIM has the capability to completely describe a robot manipulator as well as the environment with which it interacts and the objects it manipulates.

At the Air Force Institute of Technology a PUMA 560 is available for experimentation. The INITDRVR module models the base, actuator, tool, joint, and links of the robot. Swivel, hinge, and sliding joints may also be simulated. Tools are defined to grasp, hold, or carry objects, and the load objects themselves can be given mass characteristics.

The robot environment is defined by combinations of basic geometric shapes (cylinders, cones, rectangular solids, trapezoidal figures, cross-sectional beams, and special user specified objects). It would even be possible to model a pilot as a robot in ROBSIM and construct an environment duplicating any desired cockpit.

Simulation (SIMDRVR)

Defining the problem is an important step, but the actual experiment or simulation provides the data to assess the performance and potential of a robotics system design. ROBSIM has two basic types of simulation: requirements simulation and response simulation.

Requirements simulation produces information on the physical requirements to complete a set of motions. The user supplies ROBSIM with a time history file which contains information on how and where the robotics system will move objects and/or manipulator components. ROBSIM then calculates the torques and forces the actuators must supply to complete these motions. Response Simulation is the reverse of requirements simulation. Instead of providing a time history profile of motion, the user supplies a time history profile of the torques and forces applied to the robotics system. ROBSIM then calculates the resultant motion.

Requirements and response simulations have some very practical applications. They can be used to assess the peak demands on individual components of a system. This in turn helps minimize the cost of individual components. They can also assist in minimum stress path planning. If the components available cannot handle the stress applied during a given motion, it is possible that a different motion could be found to achieve the same result.

Response simulation is also used to test different control strategies. This helps eliminate inefficient or impractical control strategies and tell a designer whether at least in a computer simulation, a control strategy is workable.

Response simulation can also provide critical information on interactions with an environment. ROBSIM is able model conditions that would be very difficult to arrange experimentally and in which data collection would be very troublesome. ROBSIM is able to model the following:

1. Coulomb and static friction.
2. Contact with objects as a spring.
3. Collision impacts - modeled as an inelastic.
4. Constraints on motion (robot can't move in a given direction).
5. Disturbances - modeled as forces which act on components of the robot manipulator system.

Analysis (POSTDRVR)

ROBSIM allows the user to analyze the results of the requirements and response simulation. There are three basic analysis tools available. First, the robotics system joint motions from the simulation run and the robot tasks performed can be replayed and reviewed. Second, the resultant motion can be compared to a file of direct hardware data from an actual physical system. Third, plots of any of the translation, rotation, force or torque data computed during the requirements or response simulation can be constructed.

ROBSIM Installation

ROBSIM at AFIT operates on a VAX-11/780 with a VMS operating system. The code is written in Fortran 77 Version 4.4. The program is self-installing and assistance is available from NASA Langley if any difficulties arise.

Summary

ROBSIM has extensive capabilities as a robotics simulation program. These capabilities give ROBSIM the potential to efficiently investigate and aid in solving many of the problems facing robotics today.

Appendix F. Robot Kinematics and Inertial Navigation

The description of robot dynamics has developed from a wide variety of fields including computer graphics and mechanical engineering. Robot manipulator programming systems and languages have historically been stand-alone systems, concentrating on manipulator control and tending to ignore data manipulation (26:245). However, as robot tasks require increasing precision, data manipulation and robot characterization and motion error analysis have become imperative to further advancement of robotics.

Robot Kinematics

The robot manipulator system is defined for mathematical modeling purposes by:

(1) Links, joints, and end effectors
(2) Environment and constraints
(3) Loads and forces
(4) Tasks and goals.

Yoram Koren in his book Robotics for Engineers defines robot kinematics as follows:

> The calculation of the kinematic state of mechanical elements (i.e., their position, orientation, and rate of change) on the basis of given axial motions is referred to as the direct kinematics problem. The inverse kinematics problem refers to the calculation of required axis motions to produce desired arm motions.
>
> In the case of manipulators, direct kinematic analysis determines the state of the end effector as a function of the known states of the various joints. In addition to the state of the end effector, direct kinematic analysis also includes the

> determination of the links' position and orientation and their time derivatives. Such information is essential for any subsequent dynamic analysis or when the position and orientation of a sensor mounted on a link are required for the data processing. (17:84)

For direct kinematic analysis of more than two degrees of freedom, techniques of coordinate and vector transformations using matrices have been employed. There are several transformation techniques (17:88):

1. Complex numbers, a technique which is mainly useful in solving two-dimensional cases, but becomes complicated in three-dimensional cases (17:88).

2. Rotation matrices, a technique which is useful for vector transformations (17:88-91; 10:180).

3. The homogeneous transformation matrices technique using kinematic chains (10:179-188; 17:91-101; 27; 28).

4. The quaternion and rotation vectors. The advantages of this method are:

    (1) the definition of the relative orientation consists of three or four parameters instead of nine (3x3 matrix) as in the displacement matrix definition and

    (2) the number of the arithmetic operations required can be decreased. The inverse kinematics is implemented in the trajectory interpolator algorithm of continuous-path robots (17:117-125).

Inertial Navigation Kinematics

Inertial navigation is described in terms of reference frames, position vectors, and angular velocities (4). Both robotics and inertial navigation modeling involve the definition of coordinate frames and the use of matrix transformations of those frames to accomplish motion description. Their similarities, coupled with the very real possibility that robots will incorporate inertial navigation systems for guidance and control, encourage putting both modeling techniques into a single framework.

Unification of Notation

The fundamentals of modeling "rotation" and "translation" using dextral coordinate frames are expressed in many ways in different literature. A unifying notation is developed to facilitate comparison of robotics kinematic modeling equations with analogous classical equations used in inertial systems modeling. The effect of the associated forces and torques, which take into account the mass and inertia of the mechanical elements, is considered in dynamic rather than kinematic analysis and will therefore not be discussed here.

Notation and Tutorial

Following is a tutorial presentation of introductory robot kinematics using inertial navigation notation. A robot manipulator system is defined for mathematical modeling purposes by: (1) links, joints, and *end effectors*, (2) environment and constraints, (3) loads and forces, and (4) tasks and goals.

This tutorial will address the direct kinematics problem of computing the location in space of the end of the arm from knowing the location of all of the joints and links. The modeling of links, joints, and end effectors will be discussed using dextral (right-handed, orthogonal) coordinate frames. The rotation and translation of those frames are discussed for two-dimensional motion of a robot arm. Geometric properties of manipulators in general and the resultant system model for serial manipulators in particular will be included. This appendix will thus provide the reader with basic concepts of robot kinematics, ample references for more detailed study, and a basis for modeling the other three items.

Two coordinate and vector transformation methods will be discussed. One, the method of homogeneous transformation matrices, has been used extensively in the development of robot kinematics. The other method, that of quaternions, has not been used so extensively yet but has much potential due to the lower computational loading required for calculations.

Notation. The following notation is used in inertial navigation (4):

, => Coordinatized to a specific reference frame

/ => With respect to

| => Hold bases fixed

CAPITAL BOLD => Matrix, with the one exception of the letter **R**, which will be reserved for a position vector from the inertial frame origin O to the moving frame under consideration

- a) Transformation
- b) Coordinate change
  - 1) General: $\mathbf{T}_{b/e}$
  - 2) Dextral: $\mathbf{C}_{b/e}$ (orthonormal)

bold => Vector or Reference Frame

- a) Reference frame **b**
- b) Free Vector: $\mathbf{x}$ (coordinatized: $\mathbf{x}_{,b}$)
- c) Fixed Vector: $\mathbf{x}_{P/O}$ (coordinatized: $\mathbf{x}_{P/O,b}$)
- d) Relative to component m: $\mathbf{x}_{m,P/O,b}$

For example:

$$\mathbf{x}_{m,P/O,b} = \mathbf{C}_{b/e}\mathbf{x}_{m,P/O,e} \tag{F.1}$$

Two coordinate and vector transformation methods will be discussed. One, the method of homogeneous transformation matrices, has been used extensively in the development of robot kinematics. The other method, that of quaternions, has not been used so extensively yet but has much potential due to the lower computational loading required for calculations.

Notation. The following notation is used in inertial navigation (4):

, => Coordinatized to a specific reference frame

/ => With respect to

| => Hold bases fixed

CAPITAL BOLD => Matrix, with the one exception of the letter **R**, which will be reserved for a position vector from the inertial frame origin O to the moving frame under consideration

- a) Transformation
- b) Coordinate change
  - 1) General: $\mathbf{T}_{\mathbf{b}/\mathbf{e}}$
  - 2) Dextral: $\mathbf{C}_{\mathbf{b}/\mathbf{e}}$ (orthonormal)

bold => Vector or Reference Frame

- a) Reference frame **b**
- b) Free Vector: $\mathbf{x}$ (coordinatized: $\mathbf{x}_{,\mathbf{b}}$)
- c) Fixed Vector: $\mathbf{x}_{P/O}$ (coordinatized: $\mathbf{x}_{P/O,\mathbf{b}}$)
- d) Relative to component m: $\mathbf{x}_{m,P/O,\mathbf{b}}$

For example:

$$\mathbf{x}_{m,P/O,\mathbf{b}} = \mathbf{C}_{\mathbf{b}/\mathbf{e}}\mathbf{x}_{m,P/O,\mathbf{e}} \tag{F.1}$$

Using the above notation, the differentiation of a position vector for expressing velocity and acceleration is:

$\frac{d_e}{dt} \mathbf{R}$ => differentiation holding bases of **e** fixed

$\mathbf{R}_{|e,b}$ => above operation expressed in **b** frame

$$\mathbf{R}_{,b} \Rightarrow \frac{d}{dt} \mathbf{R}_{,bx} \mathbf{i}_b + \frac{d}{dt} \mathbf{R}_{,by} \mathbf{j}_b + \frac{d}{dt} \mathbf{R}_{,bz} \mathbf{k}_b + \mathbf{R}_{,bx} \frac{d}{dt} \mathbf{i}_b + \mathbf{R}_{,by} \frac{d}{dt} \mathbf{j}_b + \mathbf{R}_{,bz} \frac{d}{dt} \mathbf{k}_b \qquad (F.2)$$

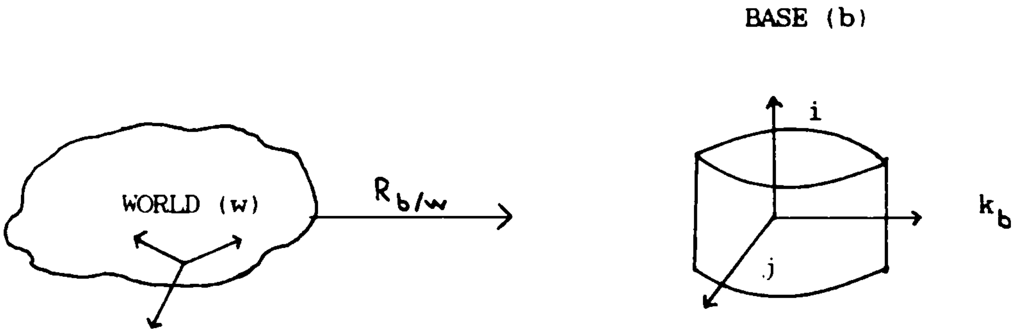
$\mathbf{w}_{b/e}$ => angular velocity of **b** frame relative to **e**

Homogeneous Transformation Matrices. A useful coordinate and vector transformation technique is based on the homogeneous transformation matrices established in 1955 by Denavit and Hartenberg (11) for use in kinematic chains.

Definition of Coordinate Frames. The first step in the transformation matrices technique is to assign coordinate systems to the moving links of the manipulator. The terms "coordinate system" and "coordinate frame" are equivalent in most robotics literature; we will use the two terms interchangeably. We will also define two other important reference frames at this point, the World coordinate frame and the Tool coordinate frame.

Link Coordinates. The right-handed coordinate frame O_i is attached to link i with the origin at joint i+1. The joints and links are typically numbered starting at the stationary base. However, for this report we will label the base coordinate system with the subscript **b** and then begin numbering consecutively from the next joint to the end of the arm.

World Coordinates. World coordinates are defined as the base reference coordinates of the robot. These coordinates are at the base joint of the robot or at a known distance from it. The World Coordinate Frame is equivalent to the Inertial Reference Frame of navigation. In inertial navigation notation, this relationship can be described by the following diagram and equation:



$$R_{b/w,w} = R_{b/w,wx}\, i_b + R_{b/w,wy} j_b + R_{b/w,wz} k_b \qquad (F.3)$$

Figure F.1. World and Base Coordinate Systems

$R_{b/w,wx}$ and ...wy and ...wz are scalars. Note that they are completely defined by the subscripts using this notation. Notice also that $R_{b/w,w}$ can be written more compactly as

$$R_{b/w,w} = C_{w/b} R_{b/w,b} \qquad (F.4)$$

using dextral coordinate change notation, where the subscripts operate as fractions do, "cancelling" to yield the vector in the desired coordinate frame. This concept is used extensively in kinematic chaining transformations.

Tool Coordinates. Tool Coordinates define robot positions relative to the end effector, or "tool." The same type of inertial navigation notation and transformation as above is used to convert from the World, or any other, frame to the Tool Coordinate.

As an example of the World and Tool Coordinate Systems, Figures 2.1 and 2.3 (from Reference 31) demonstrate the World and the Tool coordinate systems as defined for the Unimation PUMA 560 robot arm. Notice that in this particular diagram the World and the Base coordinate systems coincide. Remember that this is not always the case.

System Model for Serial Manipulator. To summarize and review the notation and definitions presented thus far, a typical basic system model for a serial manipulator will now be defined and diagrammed. This will also prepare the reader for the following sections on vector and coordinate transformations relative to robotic systems.

The system model is described by the following notation:

w -> World

b -> Base

m -> individual component (link or joint)

L -> Load (not shown in this diagram, but to be used later)

i_m, j_m, k_m or x_m, y_m, z_m	Basis definition for frame on component m
$C_{m/n}$	Dextral transformation from n into m, or of m with respect to (wrt) n
$R_{m/w}$	Position vector of m origin wrt w

r	Location of point "P" wrt base origin b, expressed in b coordinates	P/b,b

$$r_{P/w,w} = \underset{\text{defined by user}}{R_{b/w,w}} + \underset{r_{P/b,w}}{C_{w/b} r_{P/b,b}} \tag{F.5}$$

The diagram of the system model using the above notation is shown in Figure F.2.

Homogeneous Transformations. Transformation matrices define the geometric relationship between two consecutive coordinate systems as a function of the joint variable Q_i, where $Q_i = \theta_i$ for revolute joints and $Q_i = d_i$ for prismatic joints. Each transformation matrix is composed of an orientation matrix and a translation vector.

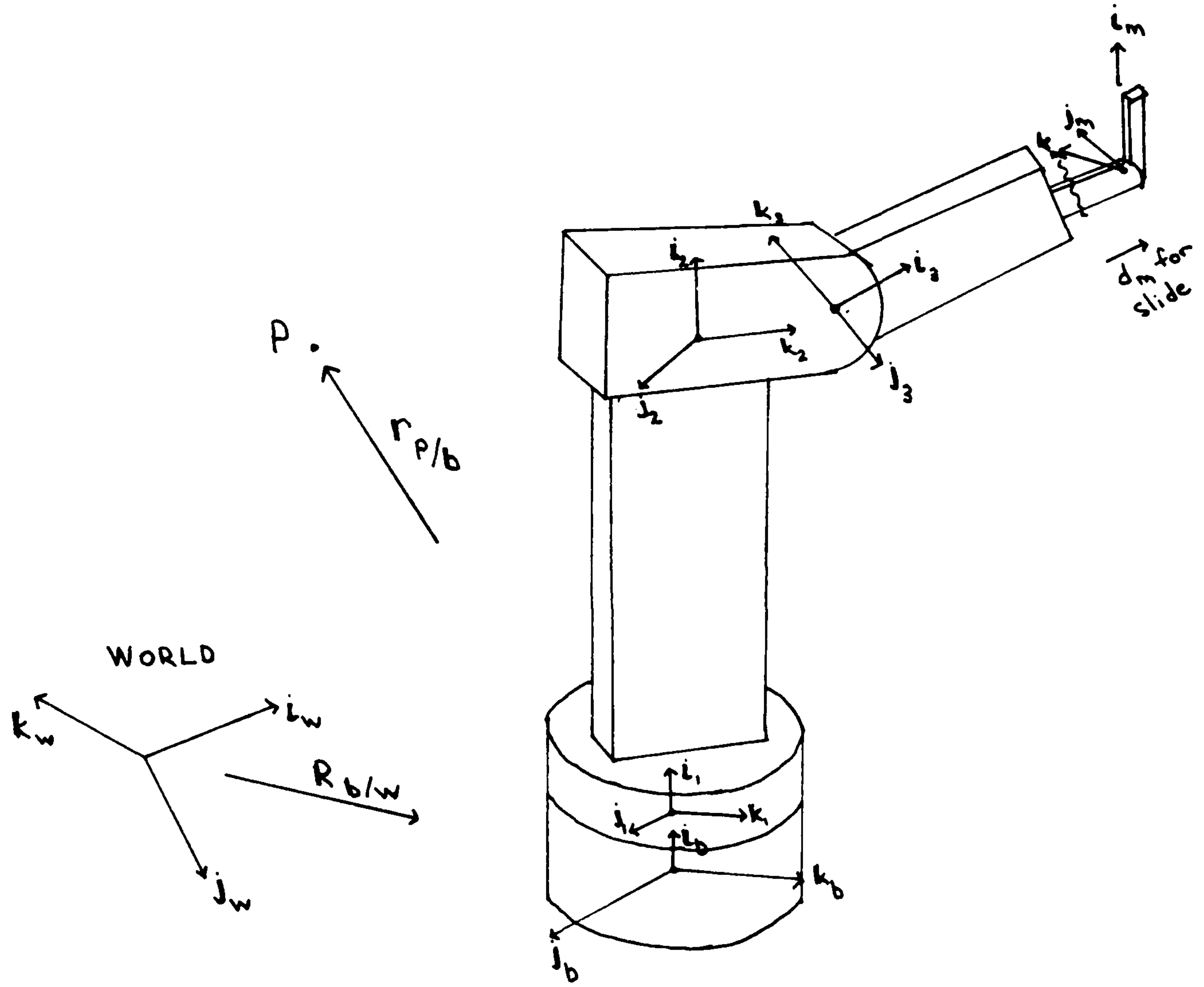


Figure F.2. Coordinate Frames of Serial Manipulator

Orientation Matrix. The orientation matrix describes the orientation of frame O_n with respect to O_m and is used to define the orientation of the link. It is the familiar "rotation matrix" or "direction cosine matrix" of dynamics (and inertial navigation) literature. A typical example is:

$$C_{m/n} = \begin{bmatrix} \cos\theta_n & -\sin\theta_n \\ \sin\theta_n & \cos\theta_n \end{bmatrix} \tag{F.6}$$

The vector components of r_m can be expressed by the vector components of r_n by: $r_m = C_{m/n} r_n$ (F.7)

Matrices can be "chained" to express successive rotations of a vector in one frame to the desired frame:

$$r_w = C_{w/b} C_{b/1} C_{1/2} r_2 \tag{F.8}$$

Translation Vector. The translation vector describes the position of the origin of frame O_n in O_m and is given by

$$d_{m/n} = \begin{bmatrix} a_n \cos\theta_n \\ a_n \sin\theta_n \end{bmatrix} \tag{F.9}$$

where a_n is the distance between O_n and O_m.

Homogeneous Transformation Matrices. The homogeneous representation of a two-dimensional vector includes three components:

$$\mathbf{r} = \begin{bmatrix} r_x \\ r_y \\ 1 \end{bmatrix} \tag{F.10}$$

The homogeneous representation of two-dimensional transformation matrices are in the form of 3x3 matrices. Three types of matrices are used:

1. The Homogeneous Rotation Matrix, which is of the form

$$\left[\begin{array}{cc:c} \multicolumn{2}{c:}{C_{m/n}} & 0 \\ & & 0 \\ \hdashline 0 & 0 & 1 \end{array}\right] \tag{F.11}$$

where C is the 2x2 orientation matrix described above. The vector $\mathbf{r}_n$ is transformed to $\mathbf{r}_m$ by Equation (F.7).

2. The Homogeneous Translation Matrix, which is of the form

$$\left[\begin{array}{cc:c} 1 & 0 & \mathbf{d}_{m/n} \\ 0 & 1 & \\ \hdashline 0 & 0 & 1 \end{array}\right] \tag{F.12}$$

where d is the translation vector given in Equation (F.9). The translation will involve the summation of a position vector, say **R**, with the translation vector of Equation (F.9).

3. The matrix of major interest in robotics, however, is the Homogeneous Displacement Matrix, which is essentially a combination of the previous two matrices. The Displacement Matrix takes on the form:

$$\left[\begin{array}{c|c} \text{Rotation Matrix} & \text{Position Vector} \\ \hline \text{Perspective Transformation} & \text{Scaling factor} \end{array}\right]$$

Figure F.3. Homogeneous Displacement Matrix

The rotation matrix, as already discussed, describes the rotation between the two coordinate systems. The position vector describes the translation, or vector distance, between the coordinate systems. Perspective transformation and scaling factor are important parameters in CAD and in graphics, where they are used extensively. In robot manipulation, the perspective transformation values are always set to 0 and the scaling factor is always 1.

The Homogeneous Displacement Matrix is also denoted as the Denavit-Hartenberg (11) matrix. The DH method, or convention, is mainly used in robot manipulators which consist of an open kinematic chain in which each joint contains one degree of freedom and the joint is either revolute or prismatic. The DH convention numbers the links and joints, establishes the links' coordinate systems, defines the joint parameters, and the forms the homogeneous displacement matrix to calculate rotation and/or translation of the manipulator. Detailed explanation and examples of the method are beyond the scope of this tutorial. The reader is referred to Denavit and Hartenberg's original paper (11), and excellent presentations are also found in References 10 and 17.

Quaternions. The relative orientation representation by the quaternion is based on the Euler theorem which states that a displacement of a rigid body with one fixed point can be described as a rotation about some axis.

A quaternion is the vector which describes that rotation. It is composed of four elements: a scalar μ which defines the magnitude of the rotation, and a three-element vector $[\alpha, \beta, \quad]^T$ which defines an axis of rotation, with α, β, and represent the equivalent Euler 3x3 matrix rotations. The 4x1 quaternion vector is then given by:

$$ {}^4q_R = \begin{bmatrix} \cos \mu/2 \\ \cos \alpha \sin \mu/2 \\ \cos \beta \sin \mu/2 \\ \cos \quad \sin \mu/2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{F.13} $$

The quaternion describing the complete rotation of a system n to system m becomes:

$$ Q_{m/n} = \begin{bmatrix} q_1 & -q_2 & -q_3 & -q_4 \\ q_2 & q_1 & q_4 & -q_3 \\ q_3 & -q_4 & q_1 & q_2 \\ q_4 & q_3 & -q_2 & q_1 \end{bmatrix} \tag{F.14} $$

Formulae exist for changing a direction cosine matrix into a quaternion transformation matrix, and vice versa. Figure F.4 depicts a quaternion rotation. More detailed explanations and examples can be found in References 17 and 21.

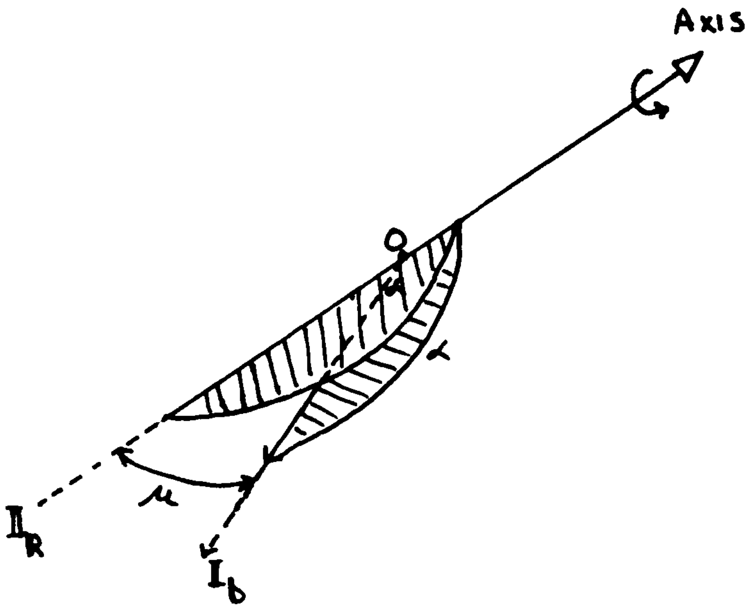


Figure F.4. Quaternion Representation

Bibliography

1. Abundis, Marcus. "Making Robots Count," *Robotics Engineering*, *8*: 17-20 (December 1986).

2. Alexander, Richard, Chief, Components Test Section (GDLC). Telephone interview. CIGTF, Holloman AFB, New Mexico, April 1987.

3. Biezad, Daniel J. Thesis Topic Proposal. AFIT/ENG, Wright-Patterson AFB, Ohio, Winter Quarter 1986.

4. Biezad, Daniel J. Class notes for EENG 534, "Instrumentation for Guidance and Control," Air Force Institute of Technology, October 1985.

5. Biezad, Daniel J., Greig, Joy Y., and Lamont, Gary B. "The Use of Computers in a Graduate Robotics Course," 1987 ASEE Annual Conference, Reno, Nevada, June 1987.

6. Black, Alan J. *A Programmer's Assistant for a Special-Purpose Dataflow Language*, MS Thesis, AFIT/GCS/ENG/85D-2, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB OH, December 1985.

7. Buchanan, John M. *Test Methods and Performance Parameters for Inertial Navigation Equipment*, MIT Final Report R-413, AF 29(600)-2068 Items 2, 3, and 6, June 1963.

8. Bustle, A., Test Program Manager (GDP). Telephone interview. Central Inertial Guidance Test Facility (CIGTF), Holloman Air Force Base, New Mexico, January 1987.

9. Cobb, Douglas. *Douglas Cobb's 1-2-3 Handbook*. New York: Bantam Books, 1986.

10. Critchlow, Arthur J. *Introduction to Robotics*. New York: Macmillan Publishing Company,1985.

11. Denavit, J., and R. S. Hartenberg. "A Kinematic Notation for Lower Pair Mechanisms Based on Matrices," *ASME Journal of Applied Mechanics*: 215-221 (June 1955).

12. *Description of Gyroscope Test Laboratory Tests and Equipment*, Working Paper WP-MDSLG-69-g, Central Inertial Guidance Test Facility, Holloman AFB, NM, September 1969.

13. "Evaluation of Automated Decisionmaking Methodologies and Development of an Integrated Robotic System Simulation," NASA Contractor Report 172401, Martin Marietta Aerospace, Denver, Colorado, September 1984.

14. "Evaluation of Automated Decisionmaking Methodologies and Development of an Integrated Robotic System Simulation," NASA Contractor Report 178050, Martin Marietta Aerospace, Denver, Colorado, March 1986.

15. Fisher, Gene H. *Cost Considerations in Systems Analysis*. New York: American Elsevier Publishing Co.,Inc., 1971.

16. Greig, Joy Y., Karnick, Drew A., Biezad, Daniel J., Lamont, Gary B. "Implementing and Testing a Robot Simulator," *Proceedings of the North Central and Indiana-Illinois Sections of the ASEE*. 404-411. Purdue University, West Lafayette, Indiana, 1986.

17. Koren, Yoram. *Robotics for Engineers*. New York: McGraw-Hill Book Company, 1984.

18. Leahy, M. B. Jr., and Saridis, G. N. "Compensation of Unmodeled PUMA Manipulator Dynamics," *Proceedings of the IEEE International Conference on Robotics and Automation*. 151-156. Raleigh, North Carolina, March 30-April 3, 1987.

19. Montalbano, George and Schlais, Richard. "Laser Guided Precision Positioning With A Gantry Robot," *Robotics Engineering*, *8*: 23-25 (February 1986).

20. Moore, James M. and Hammer, William E. Jr. *An Engineer's Guide to Spreadsheets, Word Processors, and Data Base Managers*. Norcross, Georgia: Institute of Industrial Engineers, 1986.

21. Nitzan, David. "Development of Intelligent Robots: Achievements and Issues," *IEEE Journal of Robotics and Automation*, *RA-1*: 3-13 (March 1985).

22. Nof, Shimon Y., ed. *Handbook of Industrial Robotics*. New York: John Wiley & Sons, 1985.

23. O'Toole, Tom and Cetron, Marvin. "The Coming Trend of Robots," press interview, 1984.

24. Pai, A. L. and J. W. Pan. "A Computer Graphics Simulation System for Robot Manipulator Kinematics," *1985 Conference Proceedings of the Phoenix Conference on Computers and Communications*. 237-241. March 1985.

25. Palmer, Peter J. and Gianoukos, William A. *Seminar Notes, Inertial Gyro Testing*, Instrumentation Laboratory, Massachusetts Institute of Technology, November 1966.

26. Paul, Richard P. Robot Manipulators: Mathematics, Programming, and Control. Cambridge: MIT Press, 1981.

27. Paul, R.P., Shimano, B., and Mayer, G.E. "Kinematic Control Equations for Simple Manipulators," IEEE Transactions on Systems, Man, and Cybernetics, SMC-11: 449-455 (June 1981).

28. Paul, R.P., Shimano, B., and Mayer, G.E. "Differential Kinematic Control Equations for Simple Manipulators," IEEE Transactions on Systems, Man, and Cybernetics, SMC-11: 456-460 (June 1981).

29. Simmons, Bill. Personal correspondence. Rohr Industries, Inc., Riverside, California.

30. Unimate Industrial Robot User's Guide to VAL II, Part 1, Version 1.4B, May 1985.

31. Unimation. "Unimate Industrial Robots and Cartesian Coordinates," Unimation Incorporated, Danbury, Connecticut, 1983.

32. United States Air Force, Armament Division. Laboratory Tests of the Bell Model-XI Accelerometer Conducted by the Central Inertial Guidance Test Facility. Report No. AD-TR-77-50. Holloman AFB, New Mexico: CIGTF, 6585th Test Group, May 1977.

33. United States Air Force, Armament Division. Laboratory Tests of Systron-Donner 4841F Accelerometer Conducted by the Central Inertial Guidance Test Facility. Report No. AD-TR-81-13. Holloman AFB, New Mexico: CIGTF, 6585th Test Group, January 1981.

34. W. Wrigley, W. Hollister, and W. Denhard, Gyroscopic Theory, Design and Instrumentation, The M.I.T. Press, Cambridge, Massachusetts, 1969.

Vita

Joy Y. Greig was born on 29 March 1947 in Oklahoma City, Oklahoma. She graduated from high school in Guymon, Oklahoma, in 1965, and attended Oral Roberts University, from which she received the degree of Bachelor of Arts in Mathematics, with Teacher Certification, in May 1969. Upon graduation, she received a Graduate Assistantship at the University of Tulsa and completed one year of study there before accepting a teaching position at the Cascia Hall Boys' Preparatory School in Tulsa. She then went to Australia and taught mathematics and music for three years in New South Wales and Victoria.

She became an analyst at the Central Inertial Guidance Test Facility (CIGTF), Holloman Air Force Base, New Mexico in April 1982, analyzing test data for error characterization of inertial sensors, until entering the School of Engineering, Air Force Institute of Technology, in June 1985. She will be returning to CIGTF upon graduation from AFIT.

Permanent address: 3515 Timber Dr.
Amarillo, Texas 79121

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

A182591

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION: Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release;
distribution unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S): AFIT/GE/ENG/87J

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION: School of Engineering

6b. OFFICE SYMBOL *(If applicable)*: AFIT/ENG

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State, and ZIP Code)*
Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433

7b. ADDRESS *(City, State, and ZIP Code)*

8a. NAME OF FUNDING / SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO	PROJECT NO	TASK NO	WORK UNIT ACCESSION NO.

11. TITLE *(Include Security Classification)*: See Box 19

12. PERSONAL AUTHOR(S): Joy Y. Greig

13a. TYPE OF REPORT: MS Thesis

13b. TIME COVERED: FROM ________ TO ______

14. DATE OF REPORT *(Year, Month, Day)*: 1987 May

15. PAGE COUNT: 88

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP
17	07	
12	09	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Inertial Measurement Units, Accelerometers, Gyroscopes, Error Analysis, Robotics, Test Equipment, Cost Analysis

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

Title: ROBOTICS APPLICATIONS FOR THE TESTING OF INERTIAL SENSORS

Thesis Chairman: Daniel J. Biezad, Colonel, USAF
Associate Professor of Electrical Engineering

Approved for public release: IAW AFR 190-1.
4 Jun 87
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (AFC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL: JOY Y. GREIG

22b. TELEPHONE *(Include Area Code)*: 513-255-2024

22c. OFFICE SYMBOL: AFIT/ENG

DD Form 1473, JUN 86 *Previous editions are obsolete*

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

END
8-87
DTIC